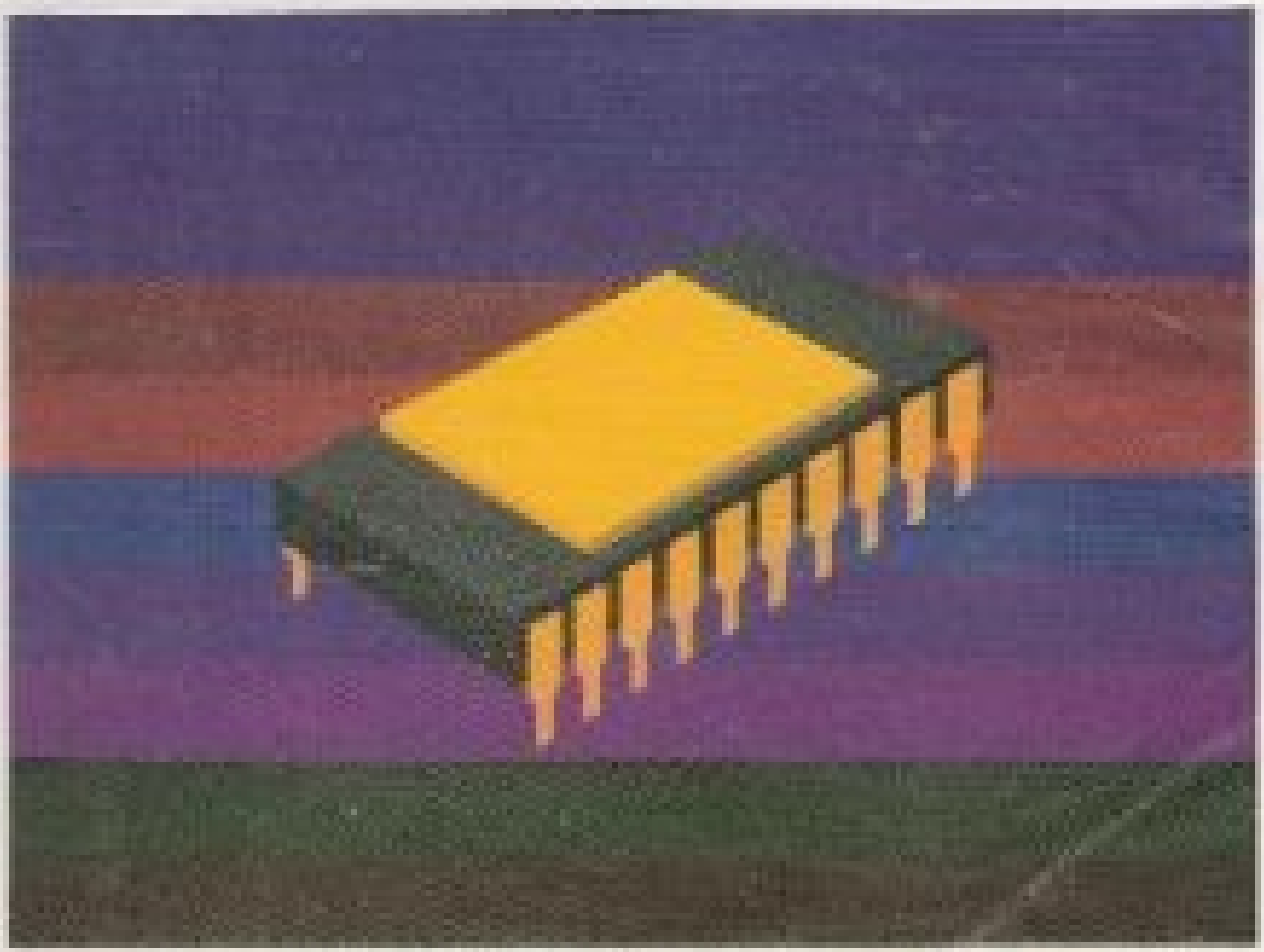
CIRCUITS NUMÉRIQUES

THÉORIE ET APPLICATIONS



RONALD J. TOCCI

Dunod

GOULET

Cette édition s'appuie sur les caractéristiques qui en ont fait un succès auprès des professeurs et des étudiants d'électronique et de technologie des systèmes ordinés.

L'ouvrage demeure un exposé complet et à jour des principes et des techniques ; il présente les familles de circuits logiques (TTL, MOS, CMOS, ECL, LOGIQUE TROIS ÉTATS) et aborde des sujets tels le codage, le multiplexage, l'échantillonnage, l'acheminement de données par bus, de même que tous les aspects standards de la logique numérique.

Cette deuxième édition a été augmentée considérablement pour refléter les développements des dernières années, notamment la montée en puissance et en capacité des différentes familles, l'arrivée en force des nouvelles mémoires à semiconducteurs et l'apparition des éléments logiques programmables qui offrent aux développeurs la souplesse nécessaire pour la mise au point de systèmes complexes.

Une dimension plus pratique a été donnée cette fois à l'ouvrage ; on y trouve désormais des sections spéciales sur le dépannage des circuits numériques.

Le livre compte plus de 200 exemples, 450 problèmes, 30 fiches techniques et un glossaire de 400 termes. Enfin, cette édition comprend un nouveau chapitre sur les microprocesseurs et le micro-ordinateur.

043364

ISBN 2-10-001576-1

Editeur : Dunod

CIRCUITS NUMERIQUES

Théorie et applications

Auteur : Ronald J. Tocci
Traduit par : André Lebel

Cet ouvrage est une version française de
Digital Systems, Principles and Applications,
6e édition, 1995, par Ronald J. Tocci,
© 1991, 1985, 1980, 1970,
par Prentice-Hall Inc., Englewood Cliffs,
New Jersey, U.S.A.

Traduction: *André Lebel*
Conception graphique de la couverture: *Denis Martineau Graphiste*
Typographie: *Typo Litho composition inc.*

© Copyright 1996, Éditions Reynald Goulet inc.

Diffusion exclusive au Canada: Éditions Reynald Goulet inc.
40, rue Mireault
Repentigny (Québec)
Canada J6A 1M1
Tél.: (514) 654-2626
Fax: (514) 654-5433
info@goulet.ca
www..goulet.ca

Diffusion exclusive en Europe: Dunod Éditeur
043364

Tous droits réservés. On ne peut reproduire aucun extrait de ce manuel par quelque procédé que ce soit, tant électronique que mécanique, en particulier par photocopie et par microfilm, sans autorisation écrite de l'éditeur. À cet effet, vous pouvez communiquer avec l'éditeur à cette adresse: Les Éditions Reynald Goulet inc., 40, rue Mireault, Repentigny (Québec) J6A 1M1.

Dépôt légal: 3e trimestre 1996
Bibliothèque nationale du Québec
Bibliothèque nationale du Canada

Imprimé au Canada

01 00 99 98 – 6 5 4 3 2

ISBN 2-89377-108-4

La *Loi sur le droit d'auteur* interdit la reproduction des oeuvres sans autorisation des titulaires de droits. Or, la photocopie non autorisée – le « photocopillage » – s'est généralisée, provoquant une baisse des achats de livres, au point que la possibilité même pour les auteurs de créer des oeuvres nouvelles et de les faire éditer par des professionnels est menacée. Nous rappelons donc que toute reproduction, partielle ou totale, du présent ouvrage est interdite sans l'autorisation écrite de l'éditeur ou d'une société de gestion dûment mandatée.

TABLE DES MATIÈRES

PRÉFACE xi

1 NOTIONS FONDAMENTALES 1

1-1 *Représentations numériques* 3
1-2 *Systèmes numériques et systèmes analogiques* 4
1-3 *Systèmes de numération* 6
1-4 *Représentation des grandeurs binaires* 10
1-5 *Circuits numériques* 11
1-6 *Transmission parallèle et transmission série* 13
1-7 *Mémoire* 14
1-8 *Calculateurs numériques* 15
Problèmes 17
Réponses aux questions de révision 18

2 SYSTÈMES DE NUMÉRATION ET CODES 19

2-1 *Conversion binaire-décimal* 21
2-2 *Conversion décimal-binaire* 21
2-3 *Système de numération octal* 22
2-4 *Système de numération hexadécimal* 25
2-5 *Code DCB (Décimal Codé Binaire)* 29
2-6 *Code majoré de trois* 31
2-7 *Code Gray* 32
2-8 *Codes alphanumériques* 33
2-9 *Détection d'erreur au moyen de la méthode de parité* 35
Problèmes 37
Réponses aux questions de révision 40

3 PORTES LOGIQUES ET ALGÈBRE BOOLÉENNE 41

3-1 *Constantes et variables booléennes* 42
3-2 *Tables de vérité* 43
3-3 *Opération OU* 44
3-4 *Opération ET* 48
3-5 *Opération NON* 51
3-6 *Mise sous forme algébrique des circuits logiques* 52
3-7 *Évaluation des sorties des circuits logiques* 53
3-8 *Matérialisation de circuits à partir d'expression booléennes* 56
3-9 *Portes NI et portes NON-ET* 57
3-10 *Théorèmes de Boole* 61
3-11 *Théorèmes de De Morgan* 65
3-12 *Universalité des portes NON-ET et des portes NI* 69
3-13 *Représentations synonymes des portes logiques* 73
3-14 *Quelle représentation de porte doit-on choisir?* 76
3-15 *Symboles logiques normalisés IEEE/ANSI* 82
Problèmes 86
Réponses aux questions de révision 92

4 CIRCUITS LOGIQUES COMBINATOIRES 93

4-1 *Somme de produits* 95
4-2 *Simplification des circuits logiques* 95
4-3 *Simplification algébrique* 96
4-4 *Conception des circuits logiques combinatoires* 100
4-5 *La méthodes des diagrammes de Karnaugh* 108
4-6 *Circuits OU exclusif et NI exclusif* 119
4-7 *Générateur et contrôleur de parité* 125
4-8 *Circuits de blocage* 127
4-9 *Caractéristiques de base des circuits numériques* 129
4-10 *Dépannage des systèmes numériques* 134
4-11 *Défaillances internes des CI numériques* 135
4-12 *Défaillances externes* 139
4-13 *Étude d'un cas de dépannage* 141
Problèmes 143
Réponses aux questions de révision 153

5 BASCULES ET ÉLÉMENTS CONNEXES 154

5-1 *Élément de mémoire en portes NON-ET* 157
5-2 *Élément de mémoire en portes NI* 163

5-3 *Étude d'un cas de dépannage* 166
5-4 *Signal d'horloge et bascules synchrones* 167
5-5 *Bascule S-C synchrone* 170
5-6 *Bascule J-K synchrone* 173
5-7 *Bascule D synchrone* 176
5-8 *Élément de mémoire D* 178
5-9 *Entrées asynchrones* 180
5-10 *Symboles IEEE/ANSI* 183
5-11 *Considérations sur la synchronisation des bascules* 186
5-12 *Problème de synchronisation possible dans les circuits à bascules* 189
5-13 *Bascule maître/esclave* 191
5-14 *Applications des bascules* 191
5-15 *Synchronisation des bascules* 191
5-16 *Détection d'une séquence d'entrée* 193
5-17 *Stockage et transfert de données* 193
5-18 *Transfert en série: registres à décalage* 196
5-19 *Division de la fréquence et comptage* 199
5-20 *Application au micro-ordinateur* 203
5-21 *Dispositifs à bascule de Schmitt* 204
5-22 *Monostable* 206
5-23 *Minuterie CI 555* 212
5-24 *Minuterie 555 vue comme un monostable* 214
5-25 *Analyse de circuits séquentiels* 218
5-26 *Multivibrateurs astables* 220
5-27 *Dépannage de circuits à bascules* 222
5-28 *Récapitulation des bascules* 227
Problèmes 229
Réponses aux questions de révision 242

6 ARITHMÉTIQUE BINAIRE: OPÉRATIONS ET CIRCUITS 244

6-1 *Addition binaire* 246
6-2 *Écriture des nombres signés* 247
6-3 *Addition: complément à 2* 253
6-4 *Soustraction: complément à 2* 254
6-5 *Multiplication de nombres binaires* 256
6-6 *Division binaire* 257
6-7 *Addition en DCB* 257
6-8 *Arithmétique hexadécimale* 259
6-9 *Circuits arithmétiques* 262
6-10 *Additionneur binaire parallèle* 263

6-11 *Conception d'un additionneur complet* 265
6-12 *Additionneur parallèle complet avec ses registres* 267
6-13 *Propagation du report* 270
6-14 *Additionneur parallèle intégré* 271
6-15 *Notation complément à 2* 273
6-16 *Additionneur DCB* 276
6-17 *Multiplicateurs binaires* 280
6-18 *Fonctions de calculs complexes en CI* 284
6-19 *Symboles IEEE/ANSI* 285
6-20 *Étude d'un cas de dépannage* 285
Problèmes 287
Réponses aux questions de révision 293

7 COMPTEURS ET REGISTRES 294

7-1 *Compteurs asynchrones (à propagation)* 296
7-2 *Compteurs à MODULO* $< 2^N$ 299
7-3 *Compteurs asynchrones CI* 303
7-4 *Décompteurs asynchrones* 309
7-5 *Retard de propagation dans les compteurs asynchrones* 310
7-6 *Compteurs synchrones (parallèles)* 313
7-7 *Décompteurs et compteurs réversibles parallèles* 316
7-8 *Compteurs préréglables* 317
7-9 *Le compteur 74193 (LS193/HC193)* 319
7-10 *Complément à la notation fonctionnelle IEEE/ANSI* 324
7-11 *Décodage d'un compteur* 326
7-12 *Décodage des signaux transitoires* 330
7-13 *Compteurs DCB montés en cascade* 333
7-14 *Conception d'un compteur synchrone* 334
7-15 *Compteurs circulaires* 340
7-16 *Applications: compteur de fréquence* 344
7-17 *Applications: horloge numérique* 350
7-18 *Registres intégrés* 353
7-19 *Écriture parallèle/lecture parallèle – les puces 74174 et 74178* 353
7-20 *Écriture série/lecture série – la puce 4731B* 357
7-21 *Écriture parallèle/lecture série – la puce 74165/74LS165/74HC165* 358
7-22 *Écriture série/lecture parallèle – la puce 74164/74LC164/74HC164* 361
7-23 *Symboles IEEE/ANSI pour les registres* 362

7-24 *Dépannage* 363
Problèmes 367
Réponses aux questions de révision 381

8 FAMILLES DE CIRCUITS INTÉGRÉS LOGIQUES 383

8-1 *Terminologie des CI numériques* 385
8-2 *Famille logique TTL* 392
8-3 *Caractéristiques de la série TTL standard* 397
8-4 *Autres séries TTL* 400
8-5 *Facteurs de charge et sortance des TTL* 405
8-6 *Autres caractéristiques TTL* 411
8-7 *Sorties TTL à collecteur ouvert* 415
8-8 *TTL trois états* 422
8-9 *Famille de CI numériques ECL* 426
8-10 *Circuits intégrés numériques MOS* 429
8-11 *Transistor MOS* 430
8-12 *Circuits numériques MOS* 432
8-13 *Caractéristiques de la logique MOS* 434
8-14 *Logique MOS complémentaire* 436
8-15 *Caractéristiques des séries CMOS* 439
8-16 *CMOS à drain ouvert et sorties trois états* 444
8-17 *Porte de transmission CMOS (interrupteur bilatéral)* 447
8-18 *Interfaçage des CI* 449
8-19 *CMOS piloté par TTL* 451
8-20 *TTL piloté par CMOS* 452
8-21 *Dépannage* 456
Problèmes 458
Réponses aux questions de révision 471

9 CIRCUITS LOGIQUES MSI 473

9-1 *Décodeurs* 475
9-2 *Pilotes/décodeurs DCB-7 segments* 483
9-3 *Afficheurs à cristaux liquides* 485
9-4 *Codeurs* 487
9-5 *Symboles IEEE/ANSI* 493
9-6 *Dépannage* 495
9-7 *Multiplexeurs (sélecteurs de données)* 497
9-8 *Applications des multiplexeurs* 502
9-9 *Démultiplexeurs (distributeurs de données)* 508

9-10 *Symbologie IEEE/ANSI: complément* 517
9-11 *Supplément de dépannage* 518
9-12 *Comparateur de grandeurs* 521
9-13 *Convertisseurs de code* 524
9-14 *Registres trois états* 528
9-15 *Bus de données* 531
Problèmes 537
Réponses aux questions de révision 551

10 CONVERTISSEURS ANALOGIQUES-NUMÉRIQUES ET NUMÉRIQUES-ANALOGIQUES 553

10-1 *Les échanges avec un monde analogique* 554
10-2 *Conversion numérique-analogique* 556
10-3 *Circuits des convertisseurs NA* 565
10-4 *Spécifications des CNA* 570
10-5 *Applications des CNA* 572
10-6 *Dépannage des CNA* 573
10-7 *Conversion analogique-numérique* 574
10-8 *CAN rampe numérique* 576
10-9 *Acquisition des données* 580
10-10 *CAN par approximations successives* 583
10-11 *CAN parallèles* 589
10-12 *Autres méthodes de conversion analogique-numérique* 591
10-13 *Voltmètre numérique* 593
10-14 *Circuit échantillonnage et mémorisation* 596
10-15 *Multiplexage* 597
10-16 *Oscilloscope numérique à mémoire* 598
Problèmes 600
Réponses aux questions de révision 610

11 DISPOSITIFS MÉMOIRES 612

11-1 *Terminologie des mémoires* 615
11-2 *Fonctionnement d'une mémoire: généralités* 618
11-3 *Connexions UCT-mémoire* 621
11-4 *Mémoires MORTES* 623
11-5 *Architecture des mémoires MORTES* 625
11-6 *Synchronisation des mémoires MORTES* 627
11-7 *Types de mémoires MORTES* 628
11-8 *Applications des mémoires MORTES* 638

11-9 *Éléments logiques programmables* 640
11-10 *Mémoires VIVES à semiconducteurs* 649
11-11 *Architecture des mémoires VIVES* 649
11-12 *Mémoires VIVES statiques (SRAM)* 652
11-13 *Mémoires VIVES dynamiques (DRAM)* 657
11-14 *Structure et fonctionnement des mémoires VIVES dynamiques* 658
11-15 *Cycles de lecture/écriture des mémoires VIVES dynamiques* 663
11-16 *Rafraîchissement des mémoires VIVES dynamiques* 665
11-17 *Extension de la longueur du mot et de la capacité* 667
11-18 *Mémoires VIVES rémanentes* 673
11-19 *Mémoires séquentielles* 675
11-20 *Mémoires magnétiques* 680
11-21 *Dépannage des systèmes de mémoires VIVES* 684
11-22 *Test des mémoires MORTES* 692
Problèmes 693
Réponses aux questions de révision 704

12 INTRODUCTION AUX MICROPROCESSEURS ET AUX MICRO-ORDINATEURS 707

12-1 *Définition d'un ordinateur* 708
12-2 *Le raisonnement dans un ordinateur* 709
12-3 *Agent secret 89* 709
12-4 *Organisation élémentaire d'un système informatique* 711
12-5 *Éléments de base d'un micro-ordinateur* 713
12-6 *Mots machines* 716
12-7 *Mots d'instruction* 717
12-8 *Exécution d'un programme en langage machine* 20
12-9 *Structure type d'un micro-ordinateur* 724
12-10 *Opérations de lecture et d'écriture* 727
12-11 *Conclusion* 731

APPENDICE I: *Glossaire* 733
APPENDICE II: *Fiches techniques des fabricants* 744
RÉPONSES AUX PROBLÈMES 754
INDEX 767

PRÉFACE

Dès sa première édition, cet ouvrage a conquis la faveur des étudiants et professeurs et son auditoire ne cesse de s'élargir. Cette nouvelle édition met l'insistance sur le côté pratique afin d'inculquer aux étudiants une connaissance des techniques de dépannage. C'est cette nouvelle orientation que veut mettre à la disposition du monde de l'enseignement la deuxième édition française de cet ouvrage. Cependant, l'essence même du livre n'est pas touchée, et il demeure un exposé complet et à jour des principes et des techniques des circuits numériques, l'outil idéal autant pour les étudiants(tes) en électrotechnique qu'en technologies des systèmes ordinés.

L'actualisation entreprise a touché deux aspects: la présentation matérielle et l'augmentation de la matière afin de refléter les développements des dernières années et de mettre l'accent sur le côté pratique. C'est ainsi que par rapport à l'ancienne édition on trouve maintenant un exposé bien intégré au texte des symboles IEEE / ANSI; de nouvelles sections réparties sur les chapitres 4 à 11 et abordant en détails le dépannage; et dans notre couverture du dépannage, des sections spéciales sur les applications aux micro-ordinateurs. Toujours dans l'optique d'aborder l'étude des systèmes numériques de façon beaucoup plus pratique, nous avons accru le nombre d'exemples et de problèmes en fin de chapitre, pour les porter respectivement à plus de 200 et de 450. En outre, on a fait appel à deux vignettes distinctes pour désigner des problèmes: une ampoule en marge pour annoncer les problèmes les plus difficiles et une sonde pour indiquer les problèmes de dépannage. Enfin, cette nouvelle édition comprend un nouveau chapitre (*Chapitre 12: Introduction au microprocesseur et au micro-ordinateur*) et un glossaire contenant quelque 250 définitions.

Voici un résumé par chapitre des nouveautés qu'on trouvera dans cette édition actualisée:

Chapitre 2: Nouveaux exemples de conversion utilisant des systèmes de numération intermédiaires.

Chapitre 3: Chronogrammes en plus grand nombre; introduction aux symboles IEEE / ANSI.

Chapitre 4: Nouvelle section sur les circuits générateur et contrôleur de parité; introduction aux familles de circuits logiques et à leurs caractéristiques de base, nouvelle matière sur les défaillances des CI et initiation aux techniques de dépannage.

Chapitre 5: Exposé plus approfondi des bascules déclenchées par un signal d'horloge et des bascules transparentes; exposé augmenté des bascules à mémoire, des bascules de Schmitt, des multivibrateurs astables, des CI et de la minuterie 555; nouvelle matière sur les problèmes liés à une dérive de l'horloge. Introduction aux diagrammes de transition d'états. Nouvelle section sur les applications aux micro-ordinateurs.

Chapitre 6: Exposé plus approfondi de la représentation en complément à 2 et de l'arithmétique hexadécimale.

Chapitre 7: Nouvelle section facultative sur la conception des compteurs asynchrones. Utilisation intensive des diagrammes de transition d'états. Introduction au registre à décalage bidirectionnel et problèmes sur ce sujet (74194).

Chapitre 8: Nouvelle matière sur différents boîtiers de CI et sur des familles rapides 74AS, 74ALS, 74HC et 74HCT; accent porté sur la façon de trouver des renseignements dans les fiches techniques; introduction au pulseur logique et au traceur de courant comme outils de dépannage.

Chapitre 9: Nouvelle matière sur le comparateur de grandeurs numérique; nouvelle section sur les circuits convertisseurs de code; plus d'applications, exposé accru sur les signaux de bus.

Chapitre 10: Nouveau sujet sur les CAN bipolaires en réseau R / 2R, sur le convertisseur AN ACD 0804; étude plus approfondie des spécifications des convertisseurs, de l'acquisition des données et de la numérisation. Nouveaux schémas et organigrammes explicatifs sur le fonctionnement du CNA. Nouvelle section sur les méthodes de conversion AN y compris la conversion par poursuite et la conversion par double rampe. Nouvelle section sur l'oscilloscope à mémoire numérique.

Chapitre 11: Nouvelle application: le générateur de fonctions. Développement de l'exposé sur les éléments logiques programmables (PLD) englobant les éléments PROM, PAL, PLA, EPLD, ainsi que les contrôleurs de mémoire VIVE dynamique, les mémoires séquentielles, les registres FIFO, l'interface UCT-mémoire; nouvelle matière sur le dépannage des mémoires VIVES et test des mémoires MORTES; couverture plus complète des mémoires MORTES à masque et des mémoires VIVES dynamiques. Illustration de la façon de déterminer et de symboliser les conditions de fusion pour une certaine table de vérité. Nouvelle section sur les NVRAM. Traitement plus succinct des mémoires à tores magnétiques, et élaboration sur de nouveaux supports comme le disque optique et le disque compact laser.

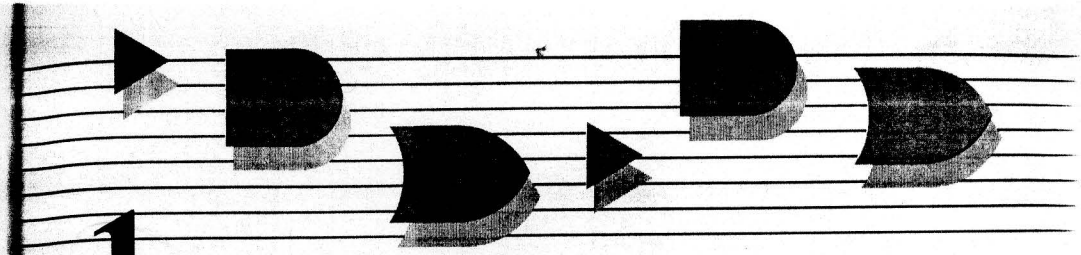
Chapitre 12: Nouveau chapitre.

CARACTÉRISTIQUES MAINTENUES

Rien de la démarche pédagogique retenue lors de la première édition n'a été changé. Le traitement de la matière continu d'être exposé en deux étapes. D'abord on explique les principes en utilisant à profusion les schémas fonctionnels sans plonger toutefois dans les détails techniques des circuits, cela afin de permettre aux étudiants(tes) de se concentrer davantage sur les principes. Ensuite, l'auteur aborde les éléments internes des CI (chapitre 8) au moment où les étudiants(tes) maîtrisent les principes des circuits logiques et sont en mesure de construire des systèmes complets, parce qu'ils comprennent bien les caractéristiques d'entrée et de sortie de chaque bloc logique.

L'exposé fait une place importante aux illustrations, aux exemples et aux questions en fin de section, questions qui incitent les étudiants(tes) à s'interroger immédiatement sur la matière présentée (*dans cette édition nous avons placé les réponses à ces questions en fin de chapitre*). Finalement, plusieurs problèmes se veulent un complément au texte, soit parce qu'ils traitent d'un circuit modifié par rapport à celui étudié, soit parce qu'ils présentent de nouvelles applications. Les étudiants(tes) ont donc la possibilité, par le biais des problèmes, d'apprendre de nouveaux aspects des systèmes numériques.

Nous espérons que cette nouvelle édition augmentée et remise à jour reflétera mieux les nouvelles réalités des systèmes numériques et continuera d'être, pour les étudiants(tes), un manuel de cours apprécié et un outil de références précieux.

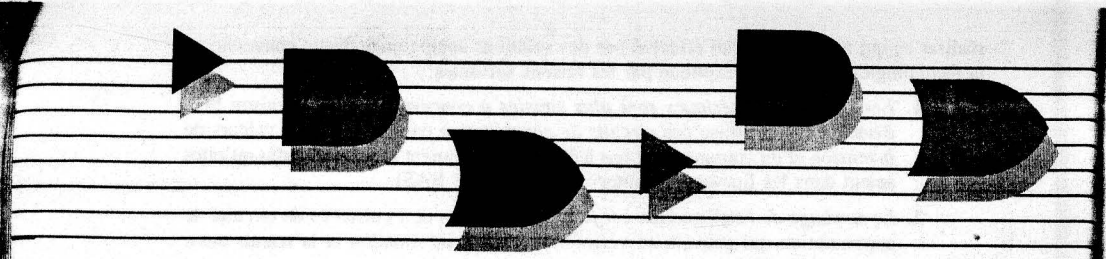


1

NOTIONS FONDAMENTALES

Contenu

1-1 Représentations numériques
1-2 Systèmes numériques et systèmes analogiques
1-3 Systèmes de numération
1-4 Représentation des grandeurs binaires
1-5 Circuits numériques
1-6 Transmission parallèle et transmission série
1-7 Mémoire
1-8 Calculateurs numériques



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Distinguer entre une représentation analogique et une représentation numérique.
- Donner les avantages et les inconvénients des systèmes analogiques, numériques et hybrides, et indiquer leurs grandes différences.
- Comprendre la raison d'être des convertisseurs analogiques-numériques (CAN) et des convertisseurs numériques-analogiques (CNA).
- Convertir des nombres décimaux en nombres binaires et vice versa.
- Reconnaître des signaux numériques caractéristiques.
- Donner le nom de plusieurs technologies de fabrication de circuits intégrés.
- Reconnaître un chronogramme.
- Préciser les différences entre la transmission série et la transmission parallèle.
- Désigner différents éléments de mémorisation.
- Décrire les principales composantes d'un ordinateur et comprendre leur rôle respectif.

Introduction

Pour la plupart d'entre nous, l'expression « numérique » fait automatiquement penser à « calculatrice numérique » ou à « ordinateur ». Et cela s'explique principalement par l'invasion dans le grand public d'un nombre considérable de calculatrices et de micro-ordinateurs puissants et peu coûteux. Il est important de se rendre compte que les calculatrices et les micro-ordinateurs ne forment qu'une seule des nombreuses applications des circuits numériques. Car on retrouve de tels circuits dans une vaste gamme de produits électroniques comme les jeux vidéo, les fours micro-ondes, les systèmes de régulation des automobiles, les appareils d'essai, les appareils de mesure, les générateurs et les oscilloscopes. Les techniques numériques ont également délogées des techniques analogiques plus anciennes dans nombre de produits de consommation courante comme les radios, les téléviseurs, les appareils d'enregistrement et de reproduction haute fidélité.

Dans ce livre, nous nous proposons d'étudier les principes et les techniques communs à tous les systèmes numériques, aussi bien d'un simple interrupteur marche-arrêt que d'un assemblage aussi complexe qu'un ordinateur. Si ce livre, comme nous l'espérons, atteint son but vous aurez acquis une connaissance solide du fonctionnement des systèmes numériques et serez aptes à comprendre les principes de l'analyse et du dépannage de n'importe quel système numérique.

Notre premier exposé porte sur les notions de base du fonctionnement de toute technique numérique. Ces notions seront développées dans la suite de ce texte selon les besoins. Ensuite, nous parlerons un peu de la terminologie nécessaire à la bonne compréhension de ce nouveau domaine; au fil des chapitres, nous introduirons les nouveaux termes indispensables. D'ailleurs vous trouverez à l'Appendice I un glossaire complet de tous les termes utilisés.

1-1 REPRÉSENTATIONS NUMÉRIQUES

Dans les sciences, dans les techniques, dans les affaires, et dans la plupart des domaines, nous sommes amenés à utiliser des *grandeurs*. Ces grandeurs sont mesurées, surveillées, enregistrées, transformées mathématiquement, observées et exploitées de diverses façons dans des systèmes réels différents. Il est important lorsqu'on travaille avec diverses grandeurs d'être en mesure de représenter correctement et précisément leurs valeurs. Il y a fondamentalement deux manières de représenter les valeurs numériques de grandeurs: la manière **analogique** et la manière **numérique**.

Représentation analogique Dans la *représentation analogique*, on fait correspondre à une grandeur une autre grandeur qui lui est directement proportionnelle. À titre d'illustration, considérons le tachymètre d'une automobile dans lequel la vitesse est figurée par la déviation d'une aiguille. La position angulaire de cette aiguille est proportionnelle à la vitesse de cette auto, et toute accélération ou tout ralentissement du véhicule se traduit par un déplacement correspondant de l'aiguille.

Un autre exemple est le thermostat courant, dans lequel l'inclinaison d'un bilame est proportionnelle à la température de la pièce. Quand la température change, la courbure du bilame change dans les mêmes proportions.

Enfin, comme troisième exemple de grandeur analogique, voyons le fonctionnement du microphone ordinaire. Dans ce dispositif, la tension de sortie produite est proportionnelle à l'amplitude des ondes sonores qui frappent la membrane du microphone. Les variations de cette tension sont analogues à celles du son capté.

Les grandeurs analogiques que l'on vient d'examiner possèdent une caractéristique importante: *elles varient graduellement à l'intérieur d'une gamme continue de valeurs*. La vitesse d'une automobile peut être comprise entre zéro et, mettons, 160 km/h. De même, la sortie d'un microphone peut être n'importe quelle valeur entre zéro et 10 mV (par exemple, 1 mV, 2,3724 mV, 9,9999 mV).

Représentation numérique Une grandeur que l'on représente numériquement n'est pas proportionnelle à une autre grandeur, mais est plutôt exprimée au moyen de symboles appelés *chiffres*. Voyons un exemple: soit une horloge numérique qui donne l'heure du jour au moyen de chiffres représentant les heures, les minutes et les secondes. Comme vous le savez, l'heure du jour est une variable continue, alors que les chiffres d'un affichage numérique, eux, ne varient pas continûment. Au contraire, l'heure qui est affichée progresse par pas d'une minute ou d'une seconde. En d'autres mots, la représentation numérique de l'heure du jour évolue de façon *discontinue*, contrairement à sa représentation analogique qui est continue (comme c'est la cas pour une montre bracelet classique).

La principale différence entre grandeur analogique et grandeur numérique peut s'exprimer simplement comme suit:

analogique = continu
numérique = discret (discontinu)

À cause de la nature discontinue de la représentation numérique, il n'y a pas d'incertitude lorsqu'on lit une telle grandeur; ce qui n'est pas le cas d'une grandeur analogique qui prête souvent à interprétation.

EXEMPLE 1-1

Dites si les systèmes suivants font intervenir des grandeurs analogiques ou des grandeurs numériques.

a) Commutateur à dix positions.
b) Ampèremètre.
c) Thermomètre.
d) Sable d'une plage.
e) Commande de volume d'un radio.

Solution

a) Numérique.
b) Analogique.
c) Analogique.
d) Numérique, puisqu'il ne peut y avoir qu'un nombre entier de grains de sable et aucune valeur intermédiaire.
e) Analogique.

QUESTION DE RÉVISION*

1. Décrivez brièvement la principale différence entre une grandeur analogique et une grandeur numérique.

1-2 SYSTÈMES NUMÉRIQUES ET SYSTÈMES ANALOGIQUES

Un **système numérique** est un regroupement d'éléments dont le rôle d'ensemble est de traiter des grandeurs physiques ou de l'information exprimées sous une forme numérique, c'est-à-dire sous la forme de valeurs discrètes. Le plus souvent ces systèmes sont électroniques, bien que certains soient mécaniques, magnétiques ou pneumatiques. Les systèmes numériques qui nous sont les plus familiers sont: les ordinateurs et les calculatrices, les montres numériques, les contrôleurs des feux de la circulation et les machines à écrire.

Un **système analogique**, par contre, est un regroupement d'éléments qui traite des grandeurs physiques de nature analogique. Dans un tel système, les grandeurs varient continûment à l'intérieur d'un intervalle de valeurs. Par exemple, l'amplitude du signal de sortie d'un haut-parleur de radiorécepteur peut prendre n'importe quelle valeur comprise entre zéro et une limite maximale. Voici quelques exemples de systèmes analogiques qui nous entourent: le téléphone, les enregistreurs et lecteurs de bande magnétique et le tachymètre d'une automobile.

Les atouts des techniques numériques Un nombre de plus en plus grand d'applications en électronique, et même d'autres disciplines, exploitent les techniques numériques pour

* Les réponses aux questions de révision sont groupées à la fin du chapitre.

réaliser ce qui auparavant était effectué par des solutions analogiques. Cette conversion à la technologie numérique s'explique par les raisons suivantes:

1. *Les systèmes numériques sont plus simples à concevoir.* Pour la raison bien évidente qu'on utilise des *circuits de commutation* dans lesquels les valeurs de la tension et du courant n'ont pas à être rigoureusement *exactes*; il suffit qu'elles soient dans les limites d'un intervalle (HAUT ou BAS).
2. *Le stockage de l'information est facile.* En raison de l'existence de circuits de commutation qui peuvent se verrouiller sur une information et la retenir aussi longtemps qu'il faut.
3. *La précision et l'exactitude sont accrues.* Les systèmes numériques peuvent opérer sur le nombre de chiffres nécessaires pour atteindre une certaine précision; et pour cela il suffit d'ajouter plus de circuits de commutation. Dans les systèmes analogiques, la précision ne peut généralement dépasser trois ou quatre chiffres, étant donné que la tension et le courant dépendent directement des valeurs des composants du circuit.
4. *On peut programmer leurs opérations.* Il est relativement facile de concevoir un circuit numérique pouvant être contrôlé par un ensemble d'instructions, appelé un *programme*. Il est vrai que l'on peut aussi programmer les systèmes analogiques, mais les opérations programmables sont peu variées et élémentaires.
5. *Les circuits numériques sont moins affectés par le bruit.* Les fluctuations parasites de la tension (bruit) ne sont pas aussi critiques dans les systèmes numériques, toujours pour la raison que le fonctionnement ne dépend pas d'une valeur exacte. Donc le bruit ne perturbe rien tant que l'on peut continuer à distinguer entre un état HAUT et un état BAS.
6. *On peut mettre un grand nombre de circuits numériques dans une puce de CI.* Les circuits analogiques ont profité aussi des fulgurantes percées dans le secteur de la technologie des CI, mais leur complexité relative et le recours à des éléments difficiles à intégrer économiquement (condensateurs à grande capacité, résistances de précision, bobines, transformateurs) empêchent d'atteindre un fort degré d'intégration.

Les limitations des techniques numériques Il n'y a vraiment qu'un seul inconvénient majeur qui joue contre les circuits numériques:

Le monde réel est analogique.

La majorité des grandeurs physiques sont analogiques, et souvent ce sont des grandeurs de cette nature que l'on retrouve aux entrées et aux sorties d'un système que l'on veut surveiller, commander ou régler. En voici quelques exemples: la température, la pression, la position, la vitesse, un niveau de liquide, un débit. Nous avons pris l'habitude de présenter ces grandeurs dans un habillage *numérique*, en disant par exemple que la température est 64° (ou 63,8° si on veut être plus précis), alors qu'en réalité nous effectuons une approximation numérique d'une grandeur essentiellement analogique.

Pour exploiter au mieux les techniques numériques dans le traitement d'entrées et de sorties analogiques, on doit mettre en œuvre les trois phases suivantes:

1. Traduire les signaux analogiques du « monde réel » en signaux numériques.
2. Traiter (agir sur) l'information numérique.
3. Convertir les sorties numériques en une forme analogique adaptée au monde réel.

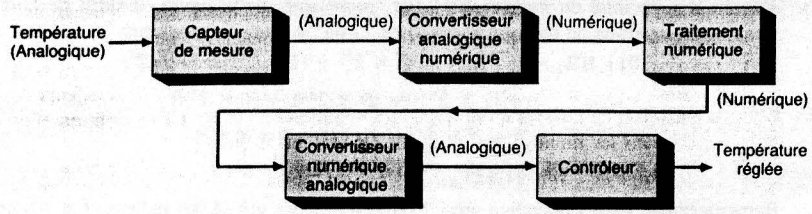


FIGURE 1-1 Schéma fonctionnel d'une chaîne de régulation de la température dans laquelle on doit effectuer des conversions numériques/analogiques pour exploiter le traitement numérique.

La figure 1-1 représente le schéma fonctionnel type d'une chaîne de régulation de la température. Comme on peut le voir, la température, une grandeur analogique, est mesurée; cette mesure est convertie en une grandeur numérique par un **convertisseur analogique-numérique (CAN)**. Cette grandeur numérique fait ensuite l'objet d'un traitement par des circuits numériques, formant ou non un calculateur numérique. La sortie numérique de ce bloc est transformée en grandeur analogique grâce à un **convertisseur numérique-analogique (CNA)**. La sortie analogique est appliquée à l'entrée d'un contrôleur qui intervient d'une certaine façon pour régler la température.

L'obligation de convertir les grandeurs numériques en grandeurs analogiques peut être vue comme un inconvénient en raison de la complexité et des déboursés accrus. En outre, un autre facteur qui a son importance est le délai qu'introduisent ces conversions. Dans nombre d'applications, ces objections sont contrebalancées par plusieurs avantages propres aux techniques numériques; c'est pourquoi la conversion entre grandeurs analogique et numérique est maintenant banale dans la technologie actuelle.

Toutefois, dans certaines situations, les techniques analogiques sont plus simples et meilleur marché. C'est le cas par exemple de l'amplification d'un signal que l'on réalise plus facilement avec des circuits analogiques.

Il devient de plus en plus habituel de retrouver un mélange de circuits numériques et de circuits analogiques dans un même système en vue de tirer profit du meilleur des deux mondes. Dans ces systèmes, dit *hybrides*, la phase de conception la plus cruciale consiste à déterminer les éléments qui seront analogiques et ceux qui seront numériques.

Finalement, on peut, sans crainte de se tromper, affirmer que l'avenir est à une implantation de plus en plus vaste des techniques numériques, en raison des économies découlant de la densité d'intégration qui devrait être encore plus intéressante dans l'avenir.

QUESTIONS DE RÉVISION

1. Indiquez les atouts des techniques numériques par rapport aux techniques analogiques.
2. Quelle est le principal obstacle à l'emploi des techniques numériques?

1-3 SYSTÈMES DE NUMÉRATION

De nombreux systèmes de numération sont utilisés en technologie numérique. Les plus courants sont les systèmes décimal, binaire, octal et hexadécimal. De toute évidence, c'est le système décimal que vous connaissez le mieux puisque vous l'utilisez tous les jours. Examinons d'abord certaines caractéristiques du système décimal qui nous aideront à mieux comprendre les autres systèmes.

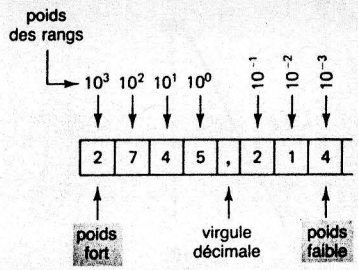


FIGURE 1-2 Les poids des rangs décimaux exprimés en puissances de 10.

Système décimal Le *système décimal* comprend *10* nombres ou symboles qui sont 0, 1, 2, 3, 4, 5, 6, 7, 8, 9; en utilisant ces symboles comme chiffres dans un nombre, on parvient à exprimer n'importe quelle grandeur. Le système décimal, appelé aussi système à *base de 10*, s'est imposé tout naturellement à l'homme puisque ce dernier possède dix doigts.

Le système décimal est dit à *poids positionnels*, en ce sens que la valeur d'un chiffre dépend de sa position (rang) dans le nombre. Par exemple, étudions le nombre décimal 453. Nous savons que le chiffre 4 correspond en réalité à 4 fois *cent*, le chiffre 5, à 5 fois *dix* et 3, à 3 fois *un*. Essentiellement, le 4 est des trois chiffres celui qui a le poids le plus élevé, on dit que c'est le *chiffre de poids le plus fort*. Par contre, le 3 est le chiffre dont le poids est le plus petit, et on dit de lui qu'il est le *chiffre de poids le plus faible*.

Voyons un autre exemple: 27,35. Ce nombre est en réalité égal à 2 fois 10, plus sept fois un, plus trois fois un dixième, plus 5 fois un centième, c'est-à-dire $2 \times 10 + 7 \times 1 + 3 \times 0{,}1 + 5 \times 0{,}01$. La virgule décimale sépare la partie entière de la partie fractionnaire du nombre.

Selon un point de vue plus rigoureux, on peut affirmer que les divers rangs possèdent des poids qui s'expriment en puissances de dix. C'est d'ailleurs ce que montre la figure 1-2, où on a représenté le chiffre 2745,214. La virgule décimale sépare les puissances de dix positives des puissances de dix négatives. Le nombre 2745,214 est donc égal à:

$$(2 \times 10^{+3}) + (7 \times 10^{+2}) + (4 \times 10^{1}) + (5 \times 10^{0}) + (2 \times 10^{-1}) + (1 \times 10^{-2}) + (4 \times 10^{-3})$$

De manière générale, tout nombre se résume à la somme des produits de chaque chiffre par le poids de son rang dans le nombre.

Comptage décimal Lorsque nous comptons dans le système décimal, nous commencons avec un 0 dans la position des unités et nous progressons en utilisant chacun des symboles jusqu'à 9. Ensuite, nous ajoutons 1 à la position immédiatement à gauche et nous recommencons avec 0 dans la position des unités (voir figure 1-3). Nous poursuivons ainsi jusqu'au nombre 99. Ensuite, nous ajoutons 1 à la troisième position et nous reprenons le comptage avec des 0 dans les deux premières positions. Nous poursuivons de cette façon jusqu'au nombre à atteindre.

Il est important de noter qu'en comptage décimal la position des unités (poids faible) prend la valeur immédiatement supérieure à chaque pas de progression du comptage, que la position des dizaines prend la valeur immédiatement supérieure à chaque dix pas de progression et que la position des centaines passe à la valeur immédiatement supérieure à chaque cent pas de progression du comptage, et ainsi de suite.

Une autre caractéristique du système décimal est de pouvoir compter 100 nombres différents = 10^2 (0 à 99)* en utilisant seulement deux positions décimales. Avec trois

* Zéro est compté comme un nombre.

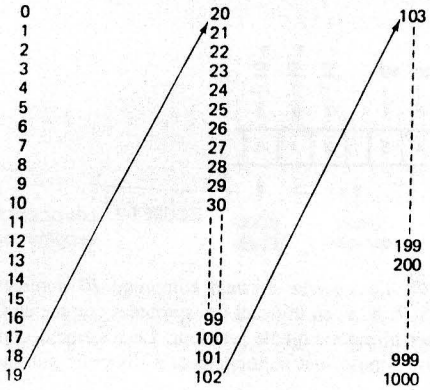


FIGURE 1-3 Suite des nombres décimaux.

positions, on peut compter 1000 nombres différents (0 à 999) etc. De manière générale, en occupant N positions, on peut créer une suite de 10^N nombres différents, en partant et en tenant compte de 0. Le nombre le plus élevé de cette suite sera $10^N - 1$.

Système binaire Malheureusement, le système décimal est difficile à adapter aux mécanismes numériques. Par exemple, il est très difficile de concevoir de l'équipement électronique qui puisse fonctionner avec dix niveaux de tension différents (un pour chaque caractère décimal 0 à 9). Par contre, il est très facile d'imaginer des systèmes électroniques simples et précis qui fonctionnent seulement avec deux niveaux de tension. C'est la raison pour laquelle la plupart des systèmes numériques ont recours au système binaire (base 2) comme système de numération de base pour leurs opérations, même si on emploie souvent d'autres systèmes de concert avec le système binaire.

Dans le *système binaire*, il n'y a que deux symboles ou chiffres possibles: 0 et 1. Malgré cela, ce système à base-2 peut représenter n'importe quelle grandeur exprimée dans le système décimal ou dans d'autres systèmes de numération. Évidemment, il faudra beaucoup plus de chiffres binaires pour exprimer une même grandeur.

Tous ce que nous venons de dire sur le système décimal s'applique intégralement au système binaire. Le système binaire est également un système à poids positionnels, puisque chaque chiffre binaire est affecté d'un poids exprimé comme une puissance de 2. C'est ce qu'on peut voir à la figure 1-4. Dans ce système, à gauche de la virgule binaire (équivalente à la virgule décimale) se trouvent les poids qui sont des puissances de 2 positives, tandis qu'à droite se trouvent les poids qui sont des puissances de 2 négatives. Ainsi pour trouver

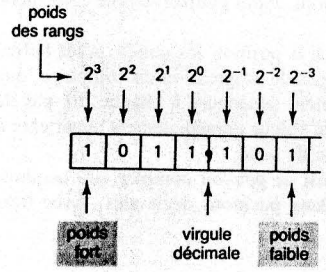


FIGURE 1-4 Les poids des rangs binaires exprimés en puissances de 2.

l'équivalent décimal du nombre 1011,101, représenté sur la figure, il suffit de faire la somme des produits de chaque chiffre (0 ou 1) par le poids de son rang.

$$1011{,}101_2 = (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) + (1 \times 2^{-1}) + (0 \times 2^{-2}) + (1 \times 2^{-3})$$
$$= 8 + 0 + 2 + 1 + 0{,}5 + 0 + 0{,}125$$
$$= 11{,}625_{10}$$

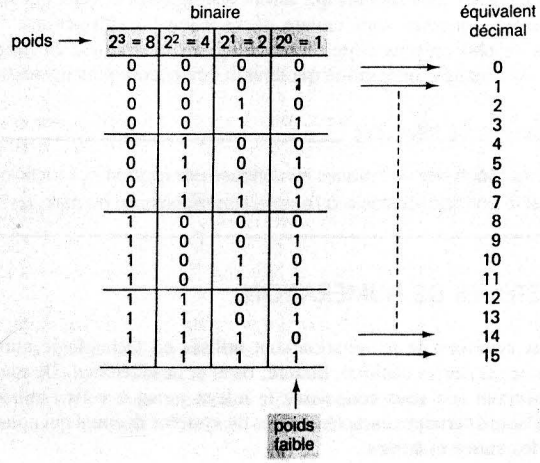
Remarquez que dans l'opération précédente, nous avons utilisé des indices (2 et 10) pour indiquer la base dans laquelle un nombre particulier est exprimé. Cette convention nous évite de confondre les nombres quand on emploie plus d'un système de numération.

Dans le système binaire, l'expression *chiffre binaire* est souvent abrégée en *bit* (contraction de binary digit), abréviation que nous utiliserons à compter de maintenant. Ainsi le nombre décrit à la figure 1-4 comprend 4 bits à gauche de la virgule binaire, qui constituent la partie entière du nombre, et 3 bits à droite de cette virgule, qui forment la partie fractionnaire du nombre. Le bit de poids le plus fort de ce nombre est celui le plus à gauche, tandis que le bit de poids le plus faible est celui le plus à droite. Ces deux bits sont indiqués sur la figure 1-4.

Comptage binaire Lorsqu'on travaille avec des nombres binaires, on cherche généralement à utiliser un nombre fixe de bits. Cette limitation est imposée par les circuits qui servent à représenter les nombres binaires. Employons des nombres binaires à 4 bits pour illustrer comment on compte en binaire.

La série de nombres (illustrée à la figure 1-5) commence avec quatre 0; c'est ce qu'on appelle le *nombre zéro*. Par la suite, à chaque fois que l'on compte 1 de plus, la valeur figurant à la position des unités (2^0) *bascule*, c'est-à-dire qu'elle passe d'une valeur binaire à l'autre. Chaque fois que cette valeur passe de 1 à 0, la valeur figurant à la position des deux (2^1) bascule. Chaque fois que la valeur de la position des deux passe de 1 à 0, la valeur figurant à la position des quatres (2^2) bascule. De la même façon, chaque fois que la valeur de la position des quatres passe de 1 à 0, la valeur de la position des huits (2^3) bascule. Le même processus s'applique aux rangs plus élevés si le nombre comporte plus de 4 bits.

FIGURE 1-5 Suite des nombres binaires.



La suite des nombres binaires manifeste une caractéristique importante, illustrée à la figure 1-5. Le bit des unités (poids faible) change de 0 à 1 ou de 1 à 0 à chaque passage au nombre immédiatement supérieur. Le deuxième bit (position des deux) demeure à 0 pendant deux nombres, puis à 1 pendant les deux nombres suivants, puis à 0 pendant les deux autres nombres, et ainsi de suite. Le troisième bit (position des quatres) demeure à 0 pendant quatre nombres, puis à 1 pendant les quatre nombres suivants, et ainsi de suite. Le quatrième bit (position des huits) reste à 0 pendant huit nombres, puis est à 1 pendant les huit nombres suivants. Si on veut compter plus loin, on doit ajouter des positions; alors on verra que cette alternance de groupes de 0 et de 1 se poursuit, la longueur de ces groupes étant 2^{N-1}. Par exemple, si on ajoute une cinquième position binaire, le cinquième bit aura la valeur 0 pendant les 16 premiers nombres, puis la valeur 1 pendant les 16 nombres suivants.

Ce qu'on a déjà expliqué pour le système décimal est également vrai pour le système binaire, c'est-à-dire qu'en utilisant N bits ou positions binaires, on peut dénombrer 2^N valeurs numériques. Par exemple, avec 2 bits, on peut compter quatre ($=2^2$) nombres (00_2 à 11_2); avec 4 bits, on peut compter 16 ($=2^4$) nombres (0000_2 à 1111_2), et ainsi de suite. Le dernier nombre est toujours celui où toutes les positions sont occupées par des 1 et sa valeur décimale est toujours $2^N - 1$. À titre d'illustration, avec 4 bits, le dernier nombre est 1111_2 dont la valeur est $2^4 - 1 = 15_{10}$.

EXEMPLE 1-2

Dites quel est le plus grand nombre que l'on peut représenter au moyen de 8 bits?

Solution: $2^N - 1 = 2^8 - 1 = 255_{10} = 11111111_2$

QUESTIONS DE RÉVISION

1. Donnez l'équivalent décimal de 1101011_2?
2. Indiquez le nombre binaire qui suit 10111_2 dans la suite des nombres binaires?
3. Quel est le plus grand nombre décimal représentable avec 12 bits?

1-4 REPRÉSENTATION DES GRANDEURS BINAIRES

Dans les systèmes numériques, l'information traitée a généralement la forme de nombres binaires. Tout dispositif ayant deux états ou conditions de fonctionnement convient à la représentation de grandeurs binaires. Par exemple, un interrupteur n'a que deux états: ouvert ou fermé. On peut décider arbitrairement qu'un interrupteur ouvert correspond au nombre binaire 0 et qu'un interrupteur fermé correspond au nombre binaire 1. Au moyen de cette convention, il nous est maintenant possible de représenter n'importe quel nombre binaire au moyen d'interrupteurs. C'est ce que montre la figure 1-6 a) sur laquelle les différents interrupteurs définissent le nombre 10010_2.
Dans l'exemple illustré à la figure 1-6 b), ce sont des trous perforés dans une bande de papier qui servent à représenter les nombres binaires. Un trou correspond à 1 et une absence de trou, à 0.

Il existe beaucoup de dispositifs ayant deux états de fonctionnement ou qui peuvent fonctionner dans deux conditions opposées. Parmi ceux-ci mentionnons: l'ampoule électrique (allumée ou éteinte), la diode (transmet ou bloque le courant), le relais (amorçé ou désamorçé), le transistor (bloqué ou saturé), la cellule photoélectrique (dans la lumière ou dans le noir), le thermostat (met en marche ou arrête), l'embrayage mécanique (embrayé ou désembrayé) et une bande magnétique (aimantée ou désaimantée).

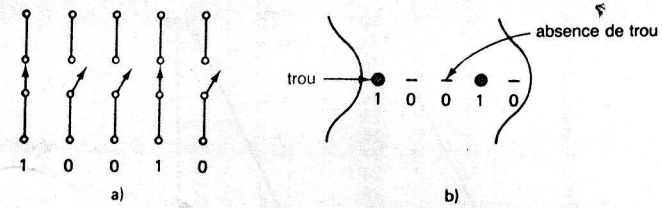


FIGURE 1-6 Représentation de nombres binaires au moyen a) d'interrupteurs; b) de trous perforés dans une bande de papier.

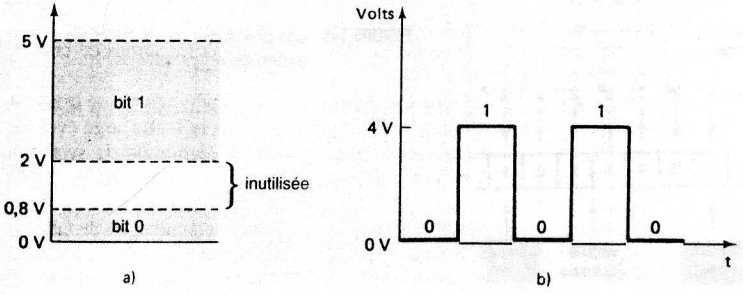
Dans l'équipement électronique numérique, l'information binaire revêt l'aspect de courants ou de tensions présents aux bornes d'entrée ou de sortie des divers circuits. Habituellement, les bits 0 et 1 correspondent à deux niveaux de tension nominaux. Par exemple, on peut établir que zéro volt représente le bit 0 et que +5 V représente le bit 1. Dans la réalité, la divergence entre les circuits fait en sorte que l'on associe à 0 et à 1 des gammes de tensions. C'est ce qu'on peut voir à la figure 1-7 a) où toute tension comprise entre 0 et 0,8 V est associée au bit 0 et toute tension comprise entre 2 et 5 V est associée au bit 1. Tous les signaux des entrées et des sorties correspondront à l'une ou à l'autre de ces gammes. La figure 1-7 b) montre une forme d'onde numérique type qui successivement définit le nombre binaire 01010.

Nous sommes maintenant en mesure de saisir une autre différence importante entre les systèmes numériques et les systèmes analogiques. Dans les systèmes numériques, la valeur exacte d'une tension n'a *pas* d'importance; par exemple, une tension de 3,6 V signifie la même chose qu'une tension de 4,3 V. Dans les systèmes analogiques, la valeur exacte d'une tension a de l'importance. Donc, si la tension analogique est proportionnelle à la température mesurée par un transducteur, 3,6 V représenterait une température différente de 4,3 V. Autrement dit, la valeur de la tension véhicule un élément informatif important. Cette caractéristique implique que la conception de circuits analogiques précis est souvent une tâche beaucoup plus difficile que la conception de circuits numériques, du fait de la fluctuation de la tension en fonction de la variation des valeurs des composants, de la température et du bruit.

1-5 CIRCUITS NUMÉRIQUES

Comme on l'a expliqué dans la dernière section, les circuits numériques sont conçus pour fournir des tensions de sortie qui se trouvent dans les gammes de tensions correspondant

FIGURE 1-7 a) Désignations habituelles des tensions dans un système numérique; b) exemple de signal numérique.



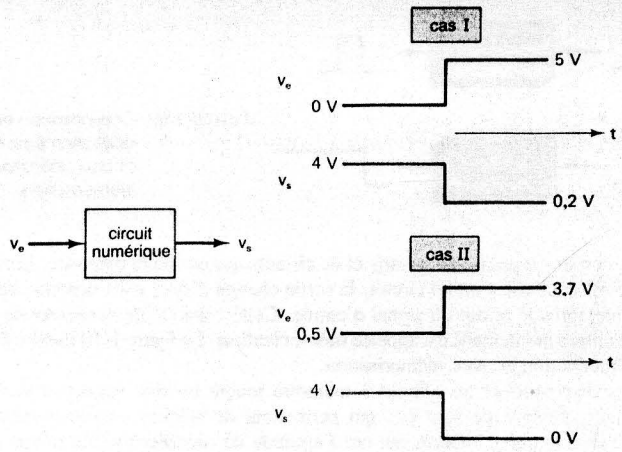


FIGURE 1-8 Le circuit numérique réagit à un niveau binaire d'entrée (0 ou 1) et non à des tensions réelles.

aux valeurs binaires 1 et 0, définies à la figure 1-7. De manière tout à fait analogue, les circuits numériques sont conçus pour réagir de façon prévisible à des tensions d'entrée qui se trouvent dans les gammes définies pour 0 et 1. Cela veut dire qu'un circuit numérique réagira toujours de la même façon à toutes les tensions d'entrée qui se situent dans la gamme de 0; de même, il ne fera pas de distinction entre les tensions comprises dans la gamme de 1.

Pour illustrer ceci, voyons la figure 1-8 où est schématisé un circuit numérique type dont l'entrée est v_e et la sortie v_s. Sur cette figure, on peut observer la sortie pour deux formes d'onde d'entrée différentes. Notez que la sortie v_s est identique dans les deux cas, puisque ces tensions d'entrée sont aux mêmes niveaux binaires bien qu'elles soient numériquement différentes.

Circuits logiques La façon suivant laquelle un circuit numérique réagit à un signal d'entrée est appelée la *logique* du circuit. Chaque type de circuit numérique obéit à un certain ensemble de règles logiques. C'est pour cette raison qu'on donne aussi aux circuits numériques le nom de *circuits logiques*. Tout au long de ce livre, nous utiliserons indistinctement ces deux expressions. Au Chapitre 3, vous pourrez vous faire une idée plus précise de ce qu'on entend par la « logique » d'un circuit.

Nous étudierons tous les types de circuits logiques employés actuellement dans les systèmes numériques. Au début, nous nous intéressons seulement à l'opération logique réalisée par ces circuits, c'est-à-dire à la relation entre les entrées et les sorties du circuit.

Circuits numériques intégrés La presque totalité des circuits numériques qui se trouvent dans l'équipement moderne d'aujourd'hui sont des circuits intégrés (CI). Le large éventail de CI logiques qui existent dans le commerce rend possible la réalisation de systèmes numériques complexes qui sont plus petits et plus fiables que leur équivalent construit avec des composants discrets.

Les circuits numériques intégrés sont fabriqués selon plusieurs technologies de fabrication de CI, les plus courantes étant les technologies TTL, CMOS, NMOS et ECL. Chacune d'elles se distingue par le type de circuit servant à réaliser l'opération logique recherchée.

Par exemple, les circuits TTL (Transistor-Transistor Logic) utilisent des transistors bipolaires comme principaux éléments de circuit, tandis que les technologies CMOS (Complementary Metal Oxide Semiconductor) emploient comme principal élément de circuit le transistor MOS à enrichissement (MOSFET). Nous aborderons l'étude des techniques de CI, leurs caractéristiques, leurs avantages et leurs inconvénients respectifs, dès que nous connaîtrons bien les circuits logiques de base.

QUESTIONS DE RÉVISION

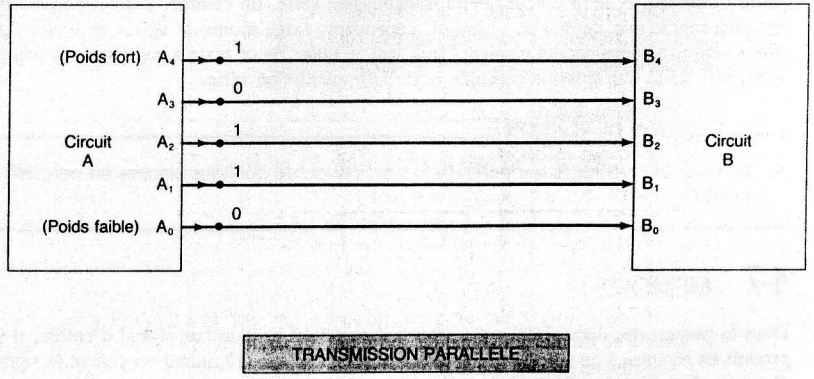
1. *Vrai ou faux:* La tension exacte d'une tension d'entrée est essentielle au fonctionnement d'un circuit numérique.
2. Est-il possible qu'un circuit numérique produise la même tension de sortie en réponse à différentes tensions d'entrée?
3. Un circuit numérique est aussi appelé circuit _____.

1-6 TRANSMISSION PARALLÈLE ET TRANSMISSION SÉRIE

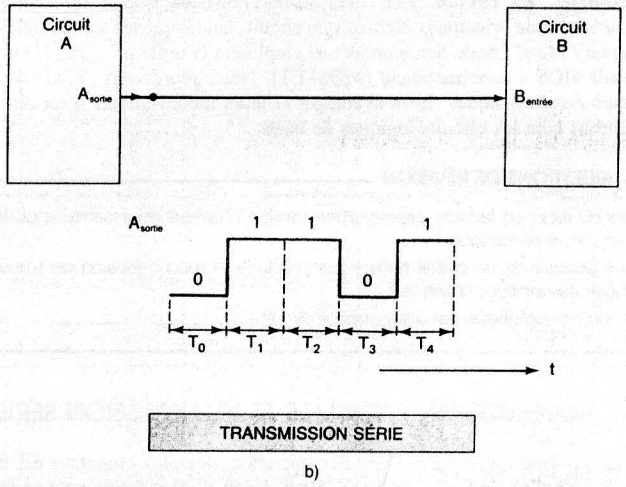
Dans un système numérique, l'une des opérations les plus courantes est la transmission d'information d'un endroit à un autre. Ce transfert d'information peut se faire entre deux éléments distant de quelques pouces sur une plaquette de circuit ou entre machines éloignées de plusieurs kilomètres, comme quand un opérateur d'ordinateur communique avec un ordinateur dans une autre ville. L'information est transmise sous forme binaire, et généralement on représente la transmission comme les tensions d'un circuit émetteur appliquées aux entrées d'un circuit récepteur. La figure 1-9 illustre les deux méthodes de base de la transmission binaire: *la transmission parallèle* et *la transmission série*.

La figure 1-9 a) montre comment le nombre 10110 est transmis en parallèle d'un circuit A à un circuit B. Chaque bit de ce nombre binaire occupe une ligne de sortie du circuit A; la sortie A_4 est le bit de poids fort et A_0 le bit de poids faible. Chaque sortie de A est connectée à l'entrée correspondante du circuit B, de sorte que les 5 bits sont acheminés simultanément (en parallèle).

FIGURE 1-9 a) Dans la transmission parallèle on utilise une ligne de connexion pour chaque bit, et tous les bits sont transmis simultanément.



a)



b)

FIGURE 1-9 (*suite*) b) Dans la transmission série, on n'utilise qu'une seule ligne de transport, et tous les bits sont émis séquentiellement (en série un à la fois).

Dans la figure 1-9 b), la transmission se fait en série et il n'y a qu'une ligne de connexion entre le circuit A et le circuit B. Ici, la sortie du circuit A est une tension qui varie à intervalles réguliers pour produire le nombre binaire à communiquer. De cette manière, la transmission se fait un bit à la fois sur la ligne. Le **chronogramme** de la figure 1-9 b) illustre la variation dans le temps du signal de sortie. Pendant le premier intervalle, T_0, le signal est au niveau zéro, pendant l'intervalle T_1, le signal est au niveau 1, et ainsi de suite.

Le principal compromis à rechercher entre transmission série et transmission parallèle consiste à choisir entre la simplicité ou la vitesse. Les données binaires sont transmises plus rapidement d'un élément à un autre d'un système binaire avec la solution en parallèle, puisque tous les bits sont envoyés en même temps et non un à la fois comme dans la solution en série. En revanche, la transmission parallèle requiert plus de lignes de connexion entre l'émetteur et le récepteur que la transmission série. En résumé, l'envoi en parallèle est plus rapide que l'envoi en série qui, cependant, exige moins de lignes de connexion. Nous serons amenés encore plusieurs fois tout au cours de ce texte à comparer les mérites respectifs de la transmission parallèle et de la transmission série.

QUESTION DE RÉVISION

1. Décrivez les avantages respectifs de la transmission de données binaires en parallèle et en série.

1-7 MÉMOIRE

Dans la plupart des dispositifs et des circuits, quand on applique un signal d'entrée, il se produit en réponse à ce dernier un signal de sortie qui disparaît quand on enlève le signal d'entrée. Les circuits réagissant ainsi n'ont pas l'aptitude de *rétention* (n'ont pas de mémoire) puisque leur sortie retourne à leur valeur normale. Dans les circuits numériques,

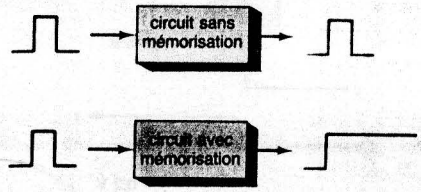


FIGURE 1-10 Comparaison entre une opération sans mémorisation et une opération avec mémorisation.

il existe certains types de dispositifs et de circuits qui ont de la mémoire. Lorsqu'un signal d'entrée est appliqué à un tel circuit, la sortie change d'état, mais demeure dans ce nouvel état même après le retrait du signal d'entrée. Cette capacité de conserver sa réponse à un signal d'entrée momentané est appelée **mémorisation**. La figure 1-10 illustre des opérations sans mémorisation et avec mémorisation.

Les dispositifs et les circuits à mémoire jouent un rôle important dans les circuits numériques, puisque ce sont eux qui permettent de stocker temporairement ou en permanence des nombres binaires qui ont l'aptitude de modifier en tout temps l'information stockée. Comme nous le verrons, les divers dispositifs de mémorisation sont, entre autres, de types magnétiques ou de types électroniques à verrouillage (*bascules*).

1-8 CALCULATEURS NUMÉRIQUES

Les techniques numériques sont exploitées dans un nombre incalculable de secteurs technologiques, mais celui des ordinateurs (**calculateurs numériques**) est de loin le secteur le plus marquant et le plus vaste. Même si les ordinateurs touchent une certaine partie de la vie de chacun de nous, il est presque certain que peu d'entre vous savent exactement ce que fait un ordinateur. De façon sommaire, on peut dire qu'un *ordinateur est un système qui effectue des opérations arithmétiques, manipule des données (le plus souvent binaires) et prend des décisions*.

La plupart des choses que font les ordinateurs peuvent être faites par des humains, mais les ordinateurs les font plus rapidement et avec plus de précision. Ils sont rapides malgré le fait que tous leurs calculs et leurs opérations sont effectués une étape à la fois. Par exemple, un être humain peut prendre une liste de dix nombres et en trouver la somme en une seule opération; il lui suffit d'aligner ces chiffres et d'additionner colonne par colonne. Un ordinateur, en revanche, ne peut additionner que deux nombres à la fois, de sorte que pour additionner la même liste de nombres il devra recourir à neuf additions. Évidemment, l'ordinateur compense largement cette lacune apparente puisqu'il ne prend qu'une microseconde ou moins pour effectuer une addition.

Un ordinateur est plus rapide et plus précis qu'une personne, mais, contrairement à cette dernière, pour qu'il fasse quelque chose, on doit lui préciser *exactement* ce qu'il doit faire à chaque étape, au moyen d'un jeu d'instructions. Ce jeu d'instructions, appelé **programme**, est rédigé par une ou plusieurs personnes et il définit chaque travail effectué par l'ordinateur. Les programmes sont placés dans l'unité de mémoire de l'ordinateur sous la forme d'un code binaire et celle-ci renferme un code exclusif correspondant à chaque instruction. L'ordinateur va chercher ces codes dans la mémoire, *un à la fois*, et exécute l'opération inscrite dans le code. Nous reviendrons abondamment sur les ordinateurs dans le reste de ce livre.

Les principaux composants d'un ordinateur Il existe plusieurs types de systèmes informatiques, mais chacun peut être divisé en quelques unités fonctionnelles fondamentales.

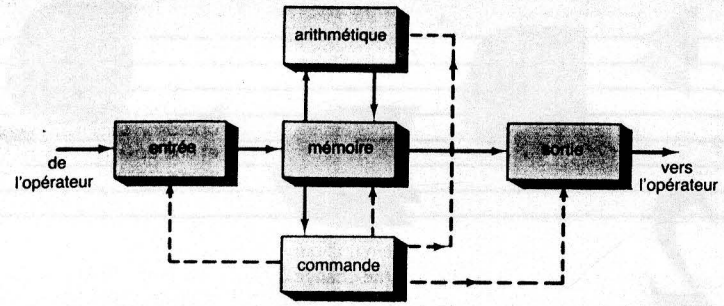


FIGURE 1-11 Schéma fonctionnel d'un ordinateur.

Chacune de ces unités exécute des fonctions particulières, et l'ensemble de toutes ces unités effectuent le travail consigné dans le programme. La figure 1-11 nous fait voir les cinq principales parties fonctionnelles, de même que l'interaction entre chacune d'elles. Les traits pleins et les flèches indiquent le déplacement d'information, alors que les lignes tiretées et les flèches représentent la transmission de signaux de synchronisation et de commande.

Les fonctions principales de chaque unité sont:

1. **Unité d'entrée**: C'est par cette unité que sont communiquées à l'ordinateur les instructions et les données que ce dernier stocke dans l'unité de mémoire jusqu'à leur utilisation. Les informations sont généralement introduites dans cette unité au moyen de cartes perforées, de bandes magnétiques, de disques magnétiques ou d'un clavier.
2. **Unité de mémoire**: La mémoire emmagasine les instructions et les données reçues de l'unité d'entrée. Elle stocke les résultats des opérations arithmétiques envoyés par l'unité arithmétique. De plus, c'est elle qui envoie l'information à l'unité de sortie.
3. **Unité de commande**: Cette unité va chercher dans la mémoire les instructions une à la fois et les interprète. Elle envoie ensuite les signaux appropriés à chacune des autres unités pour qu'une instruction particulière soit finalement exécutée.
4. **Unité arithmétique/logique**: C'est là que sont effectuées toutes les opérations arithmétiques et toutes les décisions logiques, dont les résultats sont envoyés à la mémoire.
5. **Unité de sortie**: Cette unité prélève dans la mémoire des données et les imprime, les affiche ou les fournit d'une autre manière à l'opérateur (ou au régulateur quand il s'agit d'un ordinateur de régulation de procédé).

Combien de types d'ordinateurs existe-t-il? La réponse à cette question dépend du critère qui conditionne la classification. Parmi les critères possibles, mentionnons la vitesse, la taille, la capacité de mémoire, les possibilités de traitement. Le critère de la taille étant le plus courant, bien que cela ne signifie pas toujours une capacité proportionnelle. Les trois types de base sont: le **micro-ordinateur**, le **miniordinateur** et l'**ordinateur**.

Le *micro-ordinateur* est le plus petit et le dernier-né de la famille des ordinateurs. Il est généralement formé de plusieurs puces de CI, notamment une puce **microprocesseur**, des puces mémoires et des dispositifs d'entrée et de sortie comme un clavier et un écran. Ces puces sont l'aboutissement de percées incroyables dans les techniques d'intégration à

grande échelle (LSI). Par exemple, la puce microprocesseur renferme tous les circuits de l'unité arithmétique et logique et de l'unité de commande.

Les *miniordinateurs* sont plus gros que les micro-ordinateurs et coûtent de trois à cinq fois plus cher, si on tient compte des périphériques d'entrée et de sortie. Les mini-ordinateurs sont les outils privilégiés de la régulation industrielle, des travaux scientifiques menés dans les collèges et les laboratoires de recherche et de la gestion de petites entreprises. Bien que beaucoup plus coûteux que les micro-ordinateurs, les miniordinateurs conservent de nombreux adeptes en raison de leur vitesse d'exécution généralement plus grande et de leurs capacités plus vastes. Il faut toutefois souligner que ces différences s'atténuent rapidement avec l'arrivée de nouveaux appareils.

Les *ordinateurs* les plus grands sont ceux que l'on retrouve dans les grandes entreprises, les universités et les laboratoires scientifiques. Ces « superordinateurs » peuvent coûter très cher, et ils sont associés à toute la gamme des périphériques comme les dérouleurs de bande, les unités de disques magnétiques, les perforateurs et lecteurs de cartes, les claviers, les imprimantes. Ces grands ordinateurs se chargent de longs travaux de calcul scientifiques et techniques et des traitements de gestion dans lesquels d'énormes quantités de données et d'informations sont conservées et tenues à jour.

QUESTIONS DE RÉVISION

1. Dites ce qui différencie un circuit numérique à mémorisation d'un circuit sans mémorisation.
2. Nommez les cinq principales unités fonctionnelles d'un ordinateur.
3. Nommez les trois types d'ordinateurs et dites brièvement ce qui les distingue.

PROBLÈMES

SECTION 1-2

1-1 Dites si les systèmes suivants fonctionnent avec des grandeurs analogiques ou des grandeurs numériques
   a) Atomes dans un échantillon de matériau
   b) Altimètre d'aéronef
   c) Manomètre
   d) Haut-parleur
   e) Minuterie d'un micro-ondes

SECTION 1-3

1-2 Convertissez les nombres binaires suivants en leur équivalent décimal.
   a) $11001_2 = \_\_\_\__{10}$
   b) $1001,1001_2 = \_\_\_\__{10}$
   c) $10011011001,10110 = \_\_\_\__{10}$

1-3 Utilisez 6 bits et trouvez la suite des nombres qui sépare 000000 de 111111.

1-4 Indiquez quel est le nombre maximal que l'on peut atteindre en comptant avec 10 bits.

1-5 Dites combien il faut de bits pour compter jusqu'à 511.

SECTION 1-6

1-6 Supposez que les entiers décimaux entre 0 et 15 sont représentés par des circuits numériques.

a) Dites combien il faut de circuits si on a recours à la représentation parallèle.

b) Dites combien il faut de circuits si on a recours à la représentation série.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 1-1

1. Les grandeurs analogiques prennent *n'importe quelle valeur* d'un intervalle continu; les grandeurs numériques ne peuvent avoir que des valeurs discrètes.

SECTION 1-2

1. De conception plus simples; mémorisent plus aisément des informations; sont plus précises et plus exactes; sont programmables; sont moins perturbées par le bruit; se prêtent à une grande intégration. **2.** Les grandeurs du monde réel sont analogiques.

SECTION 1-3

1. 107_{10} **2.** 11000_2 **3.** 4095_{10}

SECTION 1-5

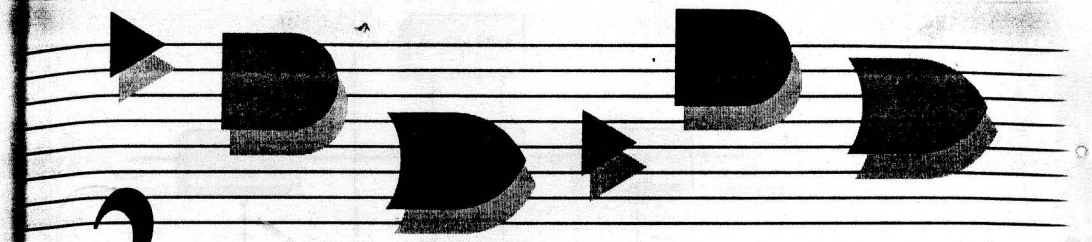
1. Faux. **2.** Oui, à la condition que les deux tensions d'entrées soient dans l'intervalle du même niveau logique. **3.** Logique.

SECTION 1-6

1. La transmission parallèle est plus rapide; la transmission série ne nécessite qu'une seule ligne.

SECTION 1-8

1. Le circuit à mémorisation en réponse à une modification momentanée de l'entrée fournit une sortie qu'il *retient*. **2.** Entrée, sortie, mémoire, arithmétique/logique, commande. **3.** Micro-ordinateur, miniordinateur, superordinateur (grand ordinateur).

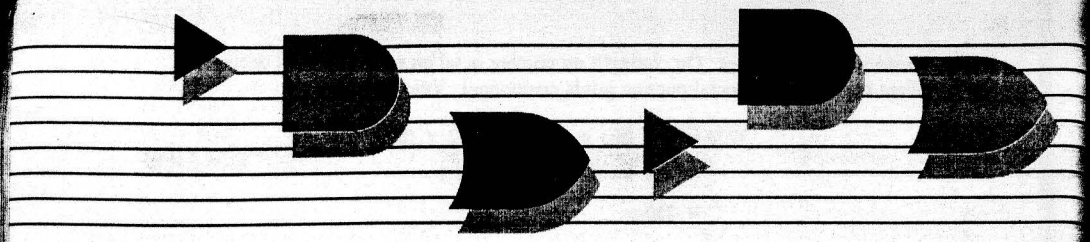


2

SYSTÈMES DE NUMÉRATION ET CODES

Contenu

2-1 Conversion binaire-décimal
2-2 Conversion décimal-binaire
2-3 Système de numération octal
2-4 Système de numération hexadécimal
2-5 Code DCB (Décimal Codé Binaire)
2-6 Code majoré de trois
2-7 Code Gray
2-8 Codes alphanumériques
2-9 Détection d'erreurs au moyen de la méthode de parité



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Réaliser selon deux méthodes différentes des conversions décimales-binaires.
- Nommer les avantages des systèmes de numération octal et hexadécimal.
- Convertir des nombres octaux et hexadécimaux en nombres décimaux ou binaires.
- Exprimer des nombres décimaux selon le code DCB.
- Comprendre la différence entre le code DCB et le code binaire pur.
- Indiquer les principales différences entre le code Gray et le code binaire.
- Apprécier l'importance des codes alphanumériques, particulièrement du code ASCII.
- Décrire la méthode de parité pour la détection des erreurs.
- Établir la parité (paire ou impaire) d'une donnée numérique.

Introduction

Le système de numération binaire est le plus important de ceux utilisés dans les circuits numériques, bien qu'il ne faille pas pour autant négliger l'importance d'autres systèmes. Le système décimal revêt de l'importance en raison de son acceptation universelle pour représenter les grandeurs du monde courant. De ce fait, il faudra parfois que des valeurs décimales soient converties en valeurs binaires avant d'être introduites dans le circuit numérique. Par exemple, lorsque vous composez un nombre décimal sur votre calculatrice (ou sur le clavier de votre ordinateur), les circuits internes convertissent ce nombre décimal en une valeur binaire.

De même, il y aura des situations où des valeurs binaires données par un circuit numérique devront être converties en valeurs décimales pour qu'on puisse les lire. Par exemple, votre calculatrice (ou votre ordinateur) calcule la réponse à un problème au moyen du système binaire puis convertit ces réponses en des valeurs décimales avant de les afficher.

Nous connaissons les systèmes binaire et décimal, étudions maintenant deux autres systèmes de numération très répandus dans les circuits numériques. Il s'agit des systèmes de numération *octal (base de 8)* et *hexadécimal (base de 16)* qui servent tous les deux au même but, soit celui de constituer un outil efficace pour représenter de gros nombres binaires. Comme nous le verrons, ces systèmes de numération ont l'avantage d'exprimer les nombres de façon que leur conversion en binaire, et vice versa, soit très facile.

Dans un système numérique, il peut arriver que trois ou quatre de ces systèmes de numération cohabitent, d'où l'importance de pouvoir convertir un système dans un autre. Le présent chapitre se propose de vous montrer comment effectuer de telles conversions. Certaines de ces conversions ne seront pas appliquées immédiatement dans l'étude des circuits numériques, mais vous constaterez au moment de l'étude des microprocesseurs que c'est une connaissance dont vous avez besoin.

Ce chapitre se veut également une introduction à certains des *codes binaires* utilisés pour représenter divers genres d'information. Ces codes agencent les 0 et le 1 de manière différente du système binaire.

2-1 CONVERSION BINAIRE-DÉCIMAL

Au Chapitre 1, nous avons dit que le système de numération binaire en est un à poids positionnels dans lequel chaque bit est affecté d'un certain poids qui dépend de son rang par rapport au bit de poids le plus faible. Ainsi tout nombre binaire peut être transformé en son équivalent décimal simplement en additionnant les poids des diverses positions où se trouve une valeur 1. Voici une illustration:

$$1 \quad 1 \quad 0 \quad 1 \quad 1 \qquad (\textit{binaire})$$

$$2^4 + 2^3 + 2^2 + 2^1 + 2^0 = 16 + 8 + 2 + 1$$

$$= 27_{10} \ (\textit{décimal})$$

Voyons un autre exemple pour un nombre ayant un plus grand nombre de bits.

$$1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1_2 =$$

$$2^7 + 0 + 2^5 + 2^4 + 0 + 2^2 + 0 + 2^0 = 181_{10}$$

Vous remarquez que la méthode consiste à trouver les poids (les puissances de 2) pour chaque position du nombre où il y a un 1, puis à additionner le tout. Remarquez que le bit de poids le plus fort a un poids de 2^7 même s'il s'agit du huitième bit; il en est ainsi parce que le bit de poids le plus faible est le premier bit et que son poids est toujours 2^0.

QUESTIONS DE RÉVISION

1. Convertissez 100011011011_2 en son équivalent décimal.
2. Dites quelle est la valeur décimale du bit de poids le plus fort d'un nombre de 16 bits?

2-2 CONVERSION DÉCIMAL-BINAIRE

Il existe deux façons de convertir un nombre décimal entier en son équivalent binaire. Une méthode qui convient bien aux petits nombres est l'inverse de la démarche suivie en 2-1. Le nombre décimal est simplement exprimé comme une somme de puissances de 2, puis on inscrit des 1 et des 0 vis-à-vis des positions binaires appropriées. Voici un exemple:

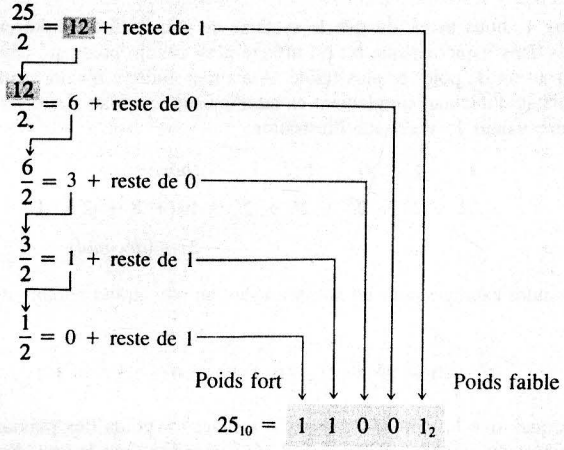
$$45_{10} = 32 + 8 + 4 + 1 = 2^5 + 0 + 2^3 + 2^2 + 0 + 2^0$$

$$= 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1_2$$

Notons qu'il y a un 0 vis-à-vis des positions 2^1 et 2^4, puisque ces positions ne sont pas utilisées pour trouver la somme en question. Voici un autre exemple:

$$76_{10} = 64 + 8 + 4 = 2^6 + 0 + 0 + 2^3 + 2^2 + 0 + 0$$
$$= 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0_2$$

Répétition de divisions L'autre méthode convient mieux aux grands nombres décimaux; il s'agit de répéter la division par 2. Cette méthode de conversion, illustrée ci-après pour le nombre 25_{10}, recourt à la répétition de la division par 2 du nombre décimal à convertir et au report des restes pour chaque division jusqu'à ce que le quotient soit 0. Notez que le nombre binaire résultant s'obtient en écrivant le premier reste à la position du bit de poids le plus faible et le dernier reste à la position du bit de poids le plus fort.

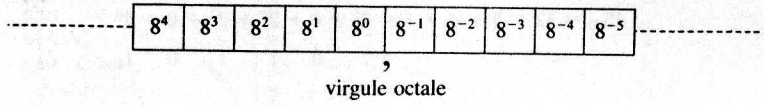


QUESTIONS DE RÉVISION

1. Convertissez 83_{10} en binaire au moyen des deux méthodes exposées.
2. Convertissez 729_{10} en binaire au moyen des deux méthodes. Contrôler votre réponse en reconvertissant en décimal.

2-3 SYSTÈME DE NUMÉRATION OCTAL

Le système de numération octal revêt une grande importance dans l'utilisation d'un ordinateur numérique. Ce système a comme base *huit*, ce qui signifie qu'il comprend huit symboles possibles, soit 0, 1, 2, 3, 4, 5, 6 et 7. Ainsi, chaque chiffre dans un nombre octal a une valeur comprise entre 0 et 7. Voici les poids de chacune des positions d'un nombre octal.



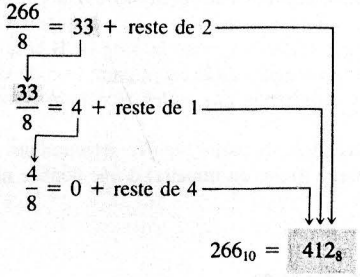
Conversion octal-décimal On convertit un nombre octal en son équivalent décimal en multipliant chaque chiffre octal par son poids positionnel. Voici un exemple:

$$372_8 = 3 \times (8^2) + 7 \times (8^1) + 2 \times (8^0)$$
$$= 3 \times 64 + 7 \times 8 + 2 \times 1$$
$$= 250_{10}$$

Autre exemple:

$$24{,}6_8 = 2 \times (8^1) + 4 \times (8^0) + 6 \times (8^{-1})$$
$$= 20{,}75_{10}$$

Conversion décimal-octal Il est possible de convertir un nombre décimal entier en son équivalent octal en employant la méthode de la répétition de divisions, la même qu'on a utilisée pour la conversion décimal-binaire, mais cette fois-ci en divisant par 8 plutôt que par 2. Voici un exemple:



Notez que le premier reste devient le chiffre de poids le plus faible du nombre octal et que le dernier reste devient le chiffre de poids le plus fort.

Si on utilise une calculatrice pour faire les divisions, on aura comme résultat un nombre avec une partie fractionnaire plutôt qu'un reste. On calcule toutefois le reste en multipliant la fraction décimale par 8. Par exemple, avec la calculatrice, la réponse de la division 266/8, est 33,25. En multipliant la partie décimale par 8, on trouve un reste de $0{,}25 \times 8 = 2$. De même, 33/8 donne 4,125, d'où un reste de $0{,}125 \times 8 = 1$.

Conversion octal-binaire Le principal avantage du système de numération octal réside dans la facilité avec laquelle il est possible de passer d'un nombre octal à un nombre binaire. Cette conversion s'effectue en transformant *chaque* chiffre du nombre octal en son équivalent binaire de trois chiffres. Voyez au tableau 2-1 les huit symboles octaux exprimés en binaire.

TABLEAU 2-1

Chiffre octal	0	1	2	3	4	5	6	7
Équivalent binaire	000	001	010	011	100	101	110	111

Au moyen de ce tableau, tout nombre octal est converti en binaire par la transformation de chacun des chiffres. Par exemple, la conversion de 472_8 va comme suit:

$$\begin{array}{ccc} 4 & 7 & 2 \\ \downarrow & \downarrow & \downarrow \\ 100 & 111 & 010 \end{array}$$

Donc le nombre octal 472_8 est équivalent au nombre binaire 100111010. Voyez un autre exemple dans lequel on convertit 5431 en binaire:

$$\begin{array}{cccc} 5 & 4 & 3 & 1 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 101 & 100 & 011 & 001 \end{array}$$

De sorte que $5431_8 = 101100011001_2$.

Conversion binaire-octal La conversion d'un nombre binaire en un nombre octal est tout simplement l'inverse de la marche à suivre précédente. Il suffit de faire avec le nombre binaire des groupes de *trois* bits en partant du chiffre de poids le plus faible, puis de convertir ces triplets en leur équivalent octal (voir tableau 2-1). À titre d'illustration, convertissons 100111010_2 en octal.

$$\begin{array}{ccc} \underbrace{1\,0\,0} & \underbrace{1\,1\,1} & \underbrace{0\,1\,0} \\ \downarrow & \downarrow & \downarrow \\ 4 & 7 & 2_8 \end{array}$$

Parfois, il arrivera que le nombre binaire ne forme pas un nombre juste de groupes de trois. Dans ce cas, on pourra ajouter un ou deux zéros à gauche du bit de poids le plus fort pour former le dernier triplet (si on lit de droite à gauche). Voici une illustration de ceci avec le nombre binaire 11010110.

$$\begin{array}{ccc} \underbrace{0\,1\,1} & \underbrace{0\,1\,0} & \underbrace{1\,1\,0} \\ \downarrow & \downarrow & \downarrow \\ 3 & 2 & 6_8 \end{array}$$

Notez l'ajout d'un zéro à gauche du bit de poids le plus fort pour obtenir un nombre juste de triplets.

Comptage en octal Le plus gros chiffre du système octal est 7, de sorte que lorsqu'on compte au moyen du système de numération octal, on commence à 0 et on progresse jusqu'à 7. Après 7, on revient à 0 et on augmente de un le chiffre immédiatement supérieur. Voici des exemples de ceci dans les suites de nombres octaux suivantes: a) 65, 66, 67, 70, 71; b) 275, 276, 277, 300.

Quand on a N chiffres octaux, on peut compter jusqu'au nombre $8^N - 1$, ce qui donne une suite de nombres de 8^N termes. Ainsi, avec trois chiffres octaux, on peut compter de 000_8 à 777_8, ce qui donne un total de $8^3 = 512$ nombres octaux différents.

Utilité du système octal La facilité avec laquelle se font les conversions entre les systèmes binaire et octal explique pourquoi le système octal est devenu une façon abrégée d'exprimer de grands nombres binaires. Dans un ordinateur, il n'est pas rare de retrouver des nombres binaires ayant jusqu'à 64 bits de longueur. Ces nombres binaires, comme nous le verrons, ne sont pas toujours des valeurs numériques, mais peuvent correspondre à un certain code

représentant des renseignements non numériques. Dans un ordinateur, un nombre binaire peut être: 1) un vrai nombre; 2) un nombre correspondant à un emplacement (adresse) en mémoire; 3) un code d'instruction; 4) un code correspondant à un caractère alphabétique ou non numérique; ou 5) un groupe de bits indiquant la situation dans laquelle se trouvent des dispositifs internes et externes de l'ordinateur.

Quand on doit travailler avec beaucoup de nombres binaires très longs, il est plus commode et plus rapide d'écrire ces nombres en octal plutôt qu'en binaire. Toutefois, ne perdez pas de vue que les circuits et les systèmes numériques fonctionnent exclusivement en binaire et que c'est par pur souci de commodité pour les opérateurs qu'on emploie la notation octale.

EXEMPLE 2-1

Convertissez 177_{10} en son équivalent binaire 8 bits en calculant d'abord sa valeur octale.

Solution

$$\frac{177}{8} = 22 + \text{reste de } 1$$

$$\frac{22}{8} = 2 + \text{reste de } 6$$

$$\frac{2}{8} = 0 + \text{reste de } 2$$

On voit que $177_{10} = 261_8$. À présent, on peut convertir facilement ce nombre octal en son équivalent binaire, soit 010110001_2, d'où l'égalité finale:

$$177_{10} = 1011001_2$$

Notez le retranchement du 0 à gauche afin d'avoir un nombre de 8 bits.

Cette méthode de conversion d'un nombre décimal en nombre octal et ensuite en nombre binaire est plus rapide que la conversion directe décimal-binaire, notamment pour les grands nombres. De la même manière, il est souvent plus rapide de convertir un nombre binaire en nombre octal et puis de transformer ce dernier en nombre décimal.

QUESTIONS DE RÉVISION

1. Convertissez 614_8 en son équivalent décimal.
2. Convertissez 146_{10} en son équivalent octal, puis passez du système octal au système binaire.
3. Convertissez 10011101_2 en son équivalent octal.
4. Écrivez les trois derniers nombres de la suite de nombres octaux suivante: 624, 625, 626.
5. Convertissez 975_{10} en binaire en passant par le système octal.
6. Convertissez le nombre binaire 1010111011 en décimal en passant par le système octal.

2-4 SYSTÈME DE NUMÉRATION HEXADÉCIMAL

Le système hexadécimal a comme base 16, ce qui implique 16 symboles de chiffres possibles, qui, dans ce cas, sont les dix chiffres 0 à 9 plus les lettres majuscules A, B, C, D, E et F. Le tableau 2-2 expose les rapports entre les systèmes hexadécimal, décimal et

binaire. Remarquez que chaque chiffre hexadécimal a comme équivalent binaire un groupe de quatre bits. Il ne faut surtout pas oublier que les chiffres hexadécimaux A à F correspondent aux valeurs décimales 10 à 15.

TABLEAU 2-2

Hexadécimal	Décimal	Binaire
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

Conversion hexadécimal-décimal Un nombre hexadécimal peut être converti en son équivalent décimal en exploitant le fait qu'à chaque position d'un chiffre hexadécimal est attribué un poids; dans ce cas-ci le nombre 16 élevé à une certaine puissance. Le chiffre de poids le plus faible a un poids de $16^0 = 1$, le chiffre immédiatement à gauche a un poids de $16^1 = 16$, l'autre chiffre immédiatement à gauche, un poids de $16^2 = 256$, et ainsi de suite. Voici un exemple sur la façon dont fonctionne ce processus de conversion.

$$\begin{aligned} 356_{16} &= 3 \times 16^2 + 5 \times 16^1 + 6 \times 16^0 \\ &= 768 + 80 + 6 \\ &= 854_{10} \end{aligned}$$

$$\begin{aligned} 2AF_{16} &= 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 \\ &= 512 + 160 + 15 \\ &= 687_{10} \end{aligned}$$

Notez dans le second exemple qu'on a substitué à A la valeur 10 et à F la valeur 15. Comme exercice, vérifiez que $1BC2_{16}$ est égal à 7106_{10}.

Conversion décimal-hexadécimal Vous vous rappelez peut-être que pour la conversion décimal-binaire nous avons eu recours à la répétition de divisions par 2, que pour la conversion décimal-octal, à la répétition de division par 8. Donc, pour convertir un nombre décimal en un nombre hexadécimal, il faut procéder de la même façon, mais cette fois en divisant par 16. Les exemples qui suivent illustrent cette technique. Remarquez comment les restes des divisions deviennent les chiffres du nombre hexadécimal; de plus, voyez comment les restes supérieurs à 9 sont exprimés au moyen des lettres A à F.

EXEMPLE 2-2

Convertissez 423_{10} en hexadécimal.

Solution

$$\frac{423}{16} = 26 + \text{reste de } 7$$

$$\frac{26}{16} = 1 + \text{reste de } 10$$

$$\frac{1}{16} = 0 + \text{reste de } 1$$

$$423_{10} = 1A7_{16}$$

EXEMPLE 2-3

Convertissez 214_{10} en hexadécimal.

Solution

$$\frac{214}{16} = 13 + \text{reste de } 6$$

$$\frac{13}{16} = 0 + \text{reste de } 13$$

$$214_{10} = D6_{16}$$

Si vous vous aidez d'une calculatrice pour faire les divisions, vous verrez sur l'affichage une partie fractionnaire plutôt qu'un reste. Pour retrouver ce reste, il vous suffit de multiplier cette partie fractionnaire par 16. À titre d'illustration, refaisons l'exemple 2-2 avec un calculatrice. Ainsi

$$\frac{214}{16} = 13{,}375$$

ce qui correspond à un reste de $0{,}375 \times 16 = 6$.

Conversion hexadécimal-binaire Comme le système de numération octal, le système de numération hexadécimal se veut une façon abrégée de représenter les nombres binaires. La conversion d'un nombre hexadécimal en un nombre binaire ne pose vraiment pas de difficulté, puisque *chaque* chiffre hexadécimal est remplacé par son équivalent binaire de 4 bits (tableau 2-2). Voici un exemple avec $9F2_{16}$.

$$\begin{aligned} 9F2_{16} &= \begin{array}{ccccccccc} 9 & & F & & 2 \\ \downarrow & & \downarrow & & \downarrow \\ 1\,0\,0\,1 & & 1\,1\,1\,1 & & 0\,0\,1\,0 \end{array} \\ &= 100111110010_2 \end{aligned}$$

Conversion binaire-hexadécimal Cette conversion est tout simplement l'inverse de la précédente. Le nombre binaire est divisé en groupes de *quatre* bits, puis on substitue à

chaque groupe son chiffre hexadécimal équivalent. Au besoin, on ajoute des zéros à gauche pour obtenir un dernier groupe de 4 bits (partie ombrée).

$$1\,1\,1\,0\,1\,0\,0\,1\,1\,0_2 = \underbrace{0\,0\,1\,1}_{3}\ \underbrace{1\,0\,1\,0}_{A}\ \underbrace{0\,1\,1\,0}_{6}$$

$$= 3A6_{16}$$

Pour passer d'un nombre hexadécimal à son équivalent binaire, il faut connaître la suite des nombres binaires de quatre bits (0000 à 1111) ainsi que le nombre correspondant en hexadécimal. Dès que cette correspondance devient un réflexe automatique, les conversions se font rapidement sans calculs. C'est ce qui explique pourquoi le système hexadécimal (et octal) est si pratique pour représenter de grands nombres binaires. Comme exercice, vérifiez que $101011111_2 = 15F_{16}$

Comptage en hexadécimal Lorsque l'on compte selon le système de numération hexadécimal, la valeur dans une position du nombre croît par pas de 1 depuis 0 jusqu'à F. Quand le chiffre dans une position est F, le chiffre suivant dans cette position est 0 et le chiffre immédiatement à gauche est augmenté de 1. C'est ce qu'on voit dans les suites de nombres hexadécimaux suivantes:

a) 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 40, 41, 42

b) 6F8, 6F9, 6FA, 6FB, 6FC, 6FD, 6FE, 6FF, 700,

Notez que le chiffre qui suit 9 dans une position est A.

EXEMPLE 2-4

Convertissez le nombre décimal 378 en un nombre binaire de 16 bits en calculant d'abord son équivalent hexadécimal.

Solution

$$\frac{378}{16} = 23 + \text{reste de } 10$$

$$\frac{23}{16} = 1 + \text{reste de } 7$$

$$\frac{1}{16} = 0 + \text{reste de } 1$$

On voit que $378_{10} = 17A_{16}$. Cette valeur hexadécimale se convertit sans difficulté en la valeur binaire 000101111010. Enfin, nous pouvons affirmer l'équivalence entre 378_{10} et un nombre de 16 bits en ajoutant à gauche quatre 0:

$$378_{10} = 0000\ \ 0001\ \ 0111\ \ 1010_2$$

EXEMPLE 2-5

Convertissez $B2F_{16}$ en octal.

Solution Il est plus facile de convertir d'abord en binaire, puis en octal.

$$\begin{aligned} B2F_{16} &= 1011\ \ 0010\ \ 111 && \{\text{conversion en binaire}\} \\ &= 101\ \ 100\ \ 101\ \ 111 && \{\text{partage en groupes de 3 bits}\} \\ &= 5\ \ 4\ \ 5\ \ 7_8 && \{\text{conversion en octal}\} \end{aligned}$$

QUESTIONS DE RÉVISION

1. Convertissez $24CE_{16}$ en décimal.
2. Convertissez 3117_{10} en hexadécimal, puis ce nombre hexadécimal en binaire.
3. Convertissez 1001011110110101_2 en hexadécimal.
4. Écrivez les quatre termes suivants de cette suite de nombres hexadécimaux: E9A, E9B, E9C, E9D, _ _ _, _ _ _, _ _ _, _ _ _.
5. Convertissez 3527_8 en hexadécimal.

2-5 CODE DCB (DÉCIMAL CODÉ BINAIRE)

L'action de faire correspondre à des nombres, des lettres ou des mots un groupe spécial de symboles s'appelle *codage* et le groupe de symboles un **code**. Un des codes que vous connaissez peut-être le mieux est le code Morse dans lequel on utilise une série de points et de traits pour représenter les lettres de l'alphabet.

Nous avons vu que tout nombre décimal pouvait être converti en son équivalent binaire. Il est possible de considérer le groupe de 0 et de 1 du nombre binaire comme un code qui représente le nombre décimal. Quand on fait correspondre à un nombre décimal son équivalent binaire, on dit qu'on fait un **codage binaire pur**.

Les circuits numériques fonctionnent avec des nombres binaires exprimés sous une forme ou sous une autre durant leurs opérations internes, malgré que le monde extérieur soit un monde décimal. Cela implique qu'il faut effectuer fréquemment des conversions entre les systèmes binaire et décimal. Nous savons que pour les grands nombres, les conversions de ce genre peuvent être longues et laborieuses. C'est la raison pour laquelle on utilise dans certaines situations un codage des nombres décimaux qui combine certaines caractéristiques du système binaire et du système décimal.

Code décimal codé binaire Si on représente *chaque* chiffre d'un nombre décimal par son équivalent binaire, on obtient le code dit **décimal codé binaire** (abrégé dans le reste du texte par DCB). Comme le plus élevé des chiffres décimaux est 9, il faut donc 4 bits pour coder les chiffres (le code binaire de 9 est 1001).

Illustrons le code DCB en prenant le nombre décimal 874 et en changeant chaque chiffre pour son équivalent binaire; cela donne:

8	7	4	décimal
↓	↓	↓	
1000	0111	0100	DCB

Comme autre nombre, codons en DCB 943

9	4	3	décimal
↓	↓	↓	
1001	0100	0011	DCB

De nouveau, on voit que chaque chiffre a été converti en son équivalent binaire pur. Notez qu'on fait *toujours* correspondre 4 bits à chaque chiffre.

Le code DCB établit donc une correspondance entre chaque chiffre d'un nombre décimal et un nombre binaire de 4 bits. Évidemment, seuls les groupes binaires 0000 à 1001 sont utilisés. Le code DCB ne fait pas usage des groupes 1010, 1011, 1100, 1101, 1110 et 1111. Autrement dit, seuls dix des 16 groupes de 4 bits sont utilisés. Si l'une des combinaisons « inadmissibles » apparaît dans une machine utilisant le code DCB, c'est généralement le signe qu'une erreur s'est produite.

EXEMPLE 2-6

Convertissez le nombre DCB 0110 1000 0011 1001 en son équivalent décimal.

Solution Fractionnez en groupes de 4 bits le nombre DCB puis procédez à la conversion.

$$\underbrace{0110}_{6}\ \underbrace{1000}_{8}\ \underbrace{0011}_{3}\ \underbrace{1001}_{9}$$

EXEMPLE 2-7

Convertissez le nombre DCB 011111000001 en son équivalent décimal.

Solution

0111 1100 0001

7 ↓ 1

groupe du code inadmissible indiquant une erreur dans le nombre DCB.

Comparaison entre code DCB et nombre binaire Il importe de bien réaliser que le code DCB n'est pas un autre système de numération comme les systèmes octal, décimal ou hexadécimal. En fait, ce code est le système décimal dont on a converti les chiffres en leur équivalent binaire. En outre, il faut bien comprendre qu'un nombre DCB n'est *pas* un nombre binaire pur. Quand on code selon le système binaire pur, on prend le nombre décimal dans son *intégralité* et on le convertit en binaire, sans le fractionner; par ailleurs, quand on code en DCB, c'est *chaque chiffre individuel* qui est remplacé par son équivalent binaire. À titre d'exemple, prenons le nombre 137 et trouvons son nombre binaire pur puis son équivalent DCB:

$$137_{10} = 10001001_2 \qquad \text{(Binaire)}$$

$$137_{10} = 0001\ \ 0011\ \ 0111_2 \qquad \text{(DCB)}$$

Le code DCB nécessite 12 bits pour représenter 137 tandis que le nombre binaire pur n'a besoin que de 8 bits. Il faut plus de bits en DCB qu'en binaire pur pour représenter les nombres décimaux de plus d'un chiffre. Comme vous le savez, il en est ainsi parce que le code DCB n'utilise pas toutes les combinaisons possibles de groupes de 4 bits; c'est donc un code peu efficace.

Le principal avantage du code DCB provient de la facilité relative avec laquelle on passe de ce code à un nombre décimal, et vice versa. Il ne faut retenir que les groupes de 4 bits des chiffres 0 à 9. C'est un avantage non négligeable du point de vue du matériel, puisque dans un système numérique ce sont des circuits logiques qui ont la charge d'effectuer ces conversions.

QUESTIONS DE RÉVISION

1. Donnez l'équivalent binaire pur du nombre décimal 178. Dans un deuxième temps, codez en DCB ce même nombre.
2. Combien faut-il de bits pour représenter un nombre décimal de 8 chiffres dans le code DCB?
3. Dites quel avantage il y a à coder un nombre décimal en DCB? Dites quel désavantage il y a.

2-6 CODE MAJORÉ DE TROIS

Le *code majoré de trois* a une certaine parenté avec le code DCB et on y a parfois recours en raison de la facilité avec laquelle on peut faire certains calculs arithmétiques. Le code majoré de trois d'un nombre décimal se trouve de la même manière que le code DCB, sauf qu'on ajoute trois à chaque chiffre décimal avant d'opérer la conversion. Par exemple, pour trouver le code majoré de trois (dit aussi code plus trois) du nombre décimal 4, on ajoute d'abord 3 à ce dernier, ce qui donne 7, puis on trouve le groupe de 4 bits qui est équivalent à ce dernier nombre, soit 0111.

Comme autre exemple, convertissons 48 en sa représentation dans le code majoré de trois.

4	8	
+ 3	+ 3	ajoutons 3 à chaque chiffre
7	11	
↓	↓	
0111	1011	convertissons en son code binaire de 4 bits

Le tableau 2-3 donne la liste des représentations DCB et code majoré de trois pour les dix chiffres décimaux. Notez que dans ces deux codes, seuls dix des 16 combinaisons possibles sont retenues. Cependant, dans le code majoré de trois, on n'utilise pas les mêmes combinaisons que dans le code DCB. Ainsi, dans le code majoré de trois, les groupes non valides sont 0000, 0001, 0010, 1101, 1110 et 1111.

TABLEAU 2-3

Décimal	DCB	Majoré de trois
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

2-7 CODE GRAY

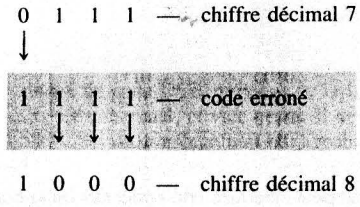
Le *code Gray* appartient à la catégorie des codes dit à *distance minimale*, du fait qu'une représentation codée ne diffère de celle qui la précède que par *un* bit. Le code Gray est un code *non pondéré*, c'est-à-dire que les positions binaires des groupes codés ne sont affectées d'aucun poids. C'est pourquoi ce code ne convient pas du tout aux calculs arithmétiques, mais se retrouve surtout dans des applications d'entrée et de sortie et dans des convertisseurs analogiques-numériques.

Le tableau 2-4 nous fait voir en parallèle les représentations du code Gray pour les nombres décimaux de 0 à 15 et celles du code binaire pur. Un examen des groupes du code Gray pour chaque chiffre décimal montre bien que lorsqu'on passe d'un *certain* chiffre au suivant, il n'y a qu'*un* bit qui est différent entre les deux groupes. Par exemple, la représentation de 3 dans le code Gray est 0010 et celle de 4 0110, il n'y a que le deuxième bit à partir de la gauche qui soit différent. De même, entre 14 et 15, soit entre 1001 et 1000, seul le dernier bit n'est pas pareil. On vient de voir là la principale caractéristique du code Gray; à l'opposé, dans le code binaire, il peut y avoir jusqu'à quatre bits différents d'une représentation à l'autre.

TABLEAU 2-4

Décimal	Binaire	Gray	Décimal	Binaire	Gray
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

Le code Gray sert souvent dans des situations où d'autres codes, comme le code binaire, peuvent produire des résultats ambigus ou erronés au moment de transitions entraînant le changement de plusieurs bits dans le code. Par exemple, dans le code binaire, lorsqu'on passe de 0111 à 1000, les quatres bits changent en même temps. Suivant le dispositif ou le circuit qui produit les bits, il pourra y avoir un différence prononcée entre les temps de transition des divers bits. Si c'est le cas, la transition 0111 à 1000 pourra occasionner un ou plusieurs états intermédiaires. Donc, s'il arrive que le bit de poids le plus fort change avant les autres, on observera les transitions suivantes:



L'apparition de 1111 est seulement temporaire mais cette représentation peut, comme en s'en doute bien, donner lieu à un fonctionnement imprévisible des circuits contrôlés par ces bits. Évidemment, avec le code Gray une telle situation est impossible puisqu'un seul bit est changé à la fois au cours d'une transition.

QUESTION DE RÉVISION

1. Donnez le principal avantage du code Gray par rapport au code binaire pur.

2-8 LES CODES ALPHANUMÉRIQUES

Un ordinateur ne serait pas d'une bien grande utilité s'il était incapable de traiter l'information non numérique. On veut dire par là qu'un ordinateur doit reconnaître des codes qui correspondent à des nombres, des lettres, des signes de ponctuation et des caractères spéciaux. Les codes de ce genre sont dit *alphanumériques*. Un ensemble de caractères complet doit renfermer les 26 lettres minuscules, les 26 lettres majuscules, les dix chiffres, les 7 signes de ponctuation et entre 20 à 40 caractères spéciaux comme +, /, #, %. On

TABLEAU 2-5 LISTE PARTIELLE DU CODE ASCII

Caractère	ASCII à 7 éléments	Octal	Hexadécimal	Caractère	ASCII À 7 éléments	Octal	Hexadécimal
A	100 0001	101	41	Y	101 1001	131	59
B	100 0010	102	42	Z	101 1010	132	5A
C	100 0011	103	43	0	011 0000	060	30
D	100 0100	104	44	1	011 0001	061	31
E	100 0101	105	45	2	011 0010	062	32
F	100 0110	106	46	3	011 0011	063	33
G	100 0111	107	47				
H	100 1000	110	48	4	011 0100	064	34
I	100 1001	111	49	5	011 0101	065	35
J	100 1010	112	4A	6	011 0110	066	36
K	100 1011	113	4B	7	011 0111	067	37
L	100 1100	114	4C	8	011 1000	070	38
				9	011 1001	071	39
M	100 1101	115	4D				
N	100 1110	116	4E	blanc	010 0000	040	20
O	100 1111	117	4F	.	010 1110	056	2E
P	101 0000	120	50	(	010 1000	050	28
Q	101 0001	121	51	+	010 1011	053	2B
R	101 0010	122	52	$	010 0100	044	24
				*	010 1010	052	2A
S	101 0011	123	53	)	010 1001	051	29
T	101 0100	124	54	–	010 1101	055	2D
U	101 0101	125	55	/	010 1111	057	2F
V	101 0110	126	56	,	010 1100	054	2C
W	101 0111	127	57	=	011 1101	075	3D
X	101 1000	130	58	(RETOUR)	000 1101	015	0D
				(INTERLIGNE)	000 1010	012	0A

peut conclure qu'un code alphanumérique reproduit tous les caractères et les diverses fonctions que l'on retrouve sur un clavier standard de machine à écrire ou d'ordinateur.

Code ASCII Le code alphanumérique le plus répandu est le code ASCII (American Standard Code for Information Interchange); on le retrouve dans la majorité des micro-ordinateurs et des miniordinateurs et dans beaucoup de gros ordinateurs. Le code ASCII (prononcé « aski ») est un code à 7 éléments, on peut donc représenter grâce à lui $2^7 = 128$ groupes codés. C'est amplement suffisant pour reproduire toutes les lettres courantes d'un clavier et les fonctions de contrôle comme (RETOUR) et (INTERLIGNE). Le tableau 2-5 contient une liste partielle du code ASCII. Dans ce dernier, en plus du groupe binaire de chaque caractère, on a donné les équivalents octaux et hexadécimaux.

EXEMPLE 2-8

Le message que voici est codé en ASCII. Que signifie-t-il?

1000001 1001001 1000100 1000101

Solution Convertissez chaque code à 7 éléments en son équivalent hexadécimal. Ce qui donne:

41 49 44 45

Maintenant localisons ces nombres hexadécimaux dans le tableau 2-5 et substituons-y les caractères associés. Nous obtenons:

A I D E

Le code ASCII sert à coder l'information alphanumérique transmise entre un ordinateur et ses périphériques d'entrée/sortie comme les écrans de visualisation ou les imprimantes. Un ordinateur recourt aussi à ce code pour stocker les informations envoyées par l'opérateur depuis son clavier. Voici un exemple qui illustre cette situation.

EXEMPLE 2-9

Un opérateur tape sur un clavier de micro-ordinateur un programme en BASIC. Cet ordinateur traduit chaque touche enfoncée en un code ASCII et conserve le tout en mémoire. Déterminez les codes qui se retrouvent en mémoire quand l'opérateur tape l'instruction en BASIC suivante:

GOTO 25

Solution Retrouvez chaque caractère de cette chaîne (y compris l'espace) dans le tableau 2-5 et consignez le code ASCII de chacun.

G	1000111
O	1001111
T	1010100
O	1001111
(espace)	0100000
2	0110010
5	0110101

QUESTIONS DE RÉVISION

1. Codez le message suivant en code ASCII en utilisant la représentation hexadécimale:
« COÛT = 72 $ »
2. Le message suivant en code ASCII est mémorisé dans des emplacements consécutifs dans l'ordinateur:
1010011 1010100 1001111 1010000
Dites quelle est la signification de ce message.

2-9 DÉTECTION D'ERREURS AU MOYEN DE LA MÉTHODE DE PARITÉ

La transmission de données binaires d'un point à un autre est monnaie courante dans les circuits numériques. D'ailleurs en voici quelques exemples:

- La transmission de la voix numérisée par des faisceaux hertziens.
- Le stockage de données dans une mémoire auxiliaire comme une bande magnétique ou un disque et leur récupération.
- La transmission d'informations d'un ordinateur à un terminal à distance ou à un autre ordinateur.

Chaque fois que de l'information transite d'un dispositif (émetteur) à un autre (récepteur), il existe une probabilité non nulle qu'une erreur survienne et que le récepteur ne reçoive pas l'information envoyée par l'émetteur. La principale raison des erreurs de transmission est le **bruit électrique**, c'est-à-dire les fluctuations de tension et de courant parasites qui se produisent dans tous les appareils électriques à des degrés divers. La figure 2-1 schématise un certain type d'erreur de transmission.

L'émetteur envoie un signal numérique presque exempt de bruit sur une voie de signal vers le récepteur. Toutefois, avant que le signal atteigne le récepteur, un certain bruit se superpose au signal original. Parfois l'amplitude de ce bruit est suffisamment grande pour changer le niveau logique du signal, comme c'est le cas pour le point *x*. Dans un tel cas, le récepteur interprète faussement ce bit en lui donnant la valeur 1, alors que l'émetteur n'a pas envoyé cette valeur.

La plupart des équipements modernes sont conçus pour être relativement sans bruit, de sorte que la probabilité d'une erreur comme celle illustrée à la figure 2-1 est très faible. Cependant, une très faible probabilité d'erreur rapportée à la transmission de centaines de milliers, voire de millions de bits par seconde (comme c'est le cas dans les systèmes numériques), signifie tout de même la présence d'erreurs qui au mieux sont désagréables, au pire peuvent être désastreuses. Pour cette raison, on implante dans nombre de systèmes numériques une certaine forme de détection des erreurs (et parfois de correction). L'une des méthodes les plus simples et les plus répandues est la détection d'erreur par la **méthode de la parité**.

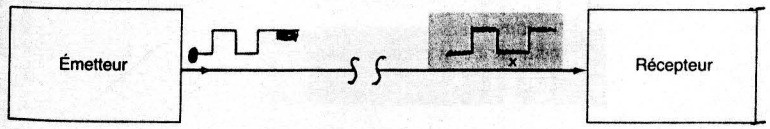


FIGURE 2-1 Comment du bruit peut introduire une erreur lors de la transmission d'une donnée numérique.

Le bit de parité Un *bit de parité* est un bit supplémentaire associé à un groupe d'un code qui doit être transféré d'un endroit à un autre. Ce bit de parité est 1 ou 0 selon le nombre de 1 qu'il y a dans le groupe. Il existe deux méthodes de contrôle au moyen de la parité.

Dans la méthode de contrôle de la *parité paire*, on fixe le bit de parité pour que le total de 1 dans la représentation codée (en tenant compte du bit de parité) soit un nombre pair. Supposons que le groupe à transmettre est 1000011, soit le code ASCII pour le caractère C. Dans ce groupe, il y a *trois* 1; il faut donc que le bit de parité soit un 1 pour que le nombre de 1 soit un nombre pair. Le nouveau code constitué par l'addition du bit de parité est:

1 1 0 0 0 0 1 1
↑ bit de parité ajouté*

Si le groupe du code renferme déjà un nombre pair de 1, le bit de parité imposé est 0. Voici un exemple: la représentation ASCII du caractère « A » est 1000001 à laquelle on attribue le bit de parité 0, de sorte que le nouveau code avec le bit de parité est 01000001.

La méthode de contrôle de la *parité impaire* est une technique tout à fait similaire à la méthode de la parité paire, sauf que dans le cas de la parité impaire, il faut que le nombre total de 1 (en comptant le bit de parité) soit un nombre *impair*. Par exemple, le groupe codé 1000001 est associé à un bit de parité de 1, tandis que le groupe 1000011 est associé à un bit de parité égal à 0.

Que l'on examine la parité paire ou la parité impaire, le bit de parité fait partie intégrante du mot codé. Ainsi, le code ASCII à 7 éléments devient un code à 8 éléments après qu'on a ajouté un bit de parité. Le bit de parité est donc un bit du code comme tous les autres.

Le rôle du bit de parité est de dépister les erreurs *monobit* qui peuvent survenir durant la transmission d'un code entre un endroit et un autre (comme entre un ordinateur et un écran de visualisation). À titre d'illustration, supposons que l'on transmette le caractère « A » et que l'on utilise la *parité impaire*. Voici le code à transmettre:

1 1 0 0 0 0 0 1

Quand le circuit récepteur reçoit ce code, il vérifie qu'il s'y trouve bien un nombre impair de 1 (en comptant le bit de parité). Si c'est le cas, ce circuit tient pour acquis que le code reçu est exact. Maintenant, supposons que des parasites ou une panne ont perturbé la transmission et que le code reçu effectivement est:

1 1 0 0 0 0 0 0

Le récepteur détermine un nombre *pair* de 1. Ceci lui indique que le code reçu est erroné, puisqu'avant la transmission l'émetteur et le récepteur s'étaient entendus pour une parité impaire. Cependant, le récepteur ne peut d'aucune manière indiquer le bit fautif, puisqu'il ignore quel code devait être transmis.

Il va de soi que la méthode de la parité n'est d'aucune utilité quand *deux* bits sont inexacts, puisque le changement de deux bits ne modifie en rien le « caractère pair ou

* Le bit de parité peut être placé au début ou à la fin du groupe, mais on préfère le placer généralement à gauche du bit de poids le plus fort.

impair » du nombre de 1 dans la représentation codée. En pratique, la méthode de parité n'est utilisée que dans les cas où la probabilité d'une erreur monobit est faible et la probabilité d'avoir deux erreurs est nulle.

Dans l'application de la méthode de la parité, l'émetteur et le récepteur se mettent d'accord, à l'avance, sur la parité surveillée: la parité paire ou la parité impaire. Aucune des parités n'est supérieure à l'autre, bien qu'on semble nettement préférer la parité paire. L'émetteur doit joindre un bit de parité approprié à chaque unité d'information qu'il expédie. Par exemple, si l'émetteur utilise des données en code ASCII, il attache un bit de parité à chaque groupe ASCII à 7 éléments. Quand le récepteur examine ce qu'il vient de recevoir de l'émetteur, il scrute chaque groupe pour voir si le nombre total de 1 (y compris le bit de parité) est conforme à l'entente passée au sujet du type de parité. On désigne souvent ce travail sous le nom de *contrôle de la parité*. Si une erreur est décelée, le récepteur peut envoyer un message à l'émetteur lui demandant de retransmettre la dernière donnée. La démarche exacte suivie lors de la détection d'une erreur est fonction de la conception particulière du système.

EXEMPLE 2-10

Un émetteur transmet une donnée en code ASCII à un récepteur avec un bit de parité paire. Montrez les codes réellement envoyés par l'émetteur dans le cas du message « ALLO ».

Solution Retrouvez les codes ASCII de chacun des caractères du message. Ensuite pour chaque code, comptez le nombre de 1. Si le nombre trouvé est pair, inscrivez un 0 comme bit de poids fort. Si le nombre est impair, inscrivez un 1 comme bit de poids fort. Les codes à 8 éléments résultants pour ce message sont ceux-ci:

bits de parité paire ajoutés ↓

A-	0	1	0	0	0	0	0	1
L-	1	1	0	0	1	1	0	0
L-	1	1	0	0	1	1	0	0
O-	1	1	0	0	1	1	1	1

QUESTIONS DE RÉVISION

1. Associez un bit de parité impaire au code ASCII du symbole «$» et exprimez la réponse en hexadécimal.
2. Associez un bit de parité paire au code DCB du nombre 69; quel est le résultat?
3. Pourquoi la méthode de parité ne peut-elle déceler une double erreur de transmission?

PROBLÈMES

SECTIONS 2-1 et 2-2

2-1 Convertissez en décimal les nombres binaires suivants:
a) 10110 b) 10001101 c) 100100001001 d) 1111010111 e) 10111111

2-2 Convertissez en binaire les nombres décimaux suivants:
a) 37 b) 14 c) 189 d) 205 e) 2313 f) 511

2-3 Indiquez le plus grand nombre décimal que l'on peut représenter avec un nombre binaire de 8 bits. Avec un nombre binaire de 16 bits.

SECTION 2-3

2-4 Trouvez l'équivalent décimal de chacun des nombres octaux que voici.
a) 743 b) 36 c) 3777 d) 257 e) 1204

2-5 Trouvez l'équivalent octal de chacun des nombres décimaux que voici.
a) 59 b) 372 c) 919 d) 65536 e) 255

2-6 Convertissez en binaire les nombres octaux du problème 2-4.

2-7 Convertissez en octal les nombres binaires du problème 2-1.

2-8 Donnez la suite des nombres octaux de 165_8 à 200_8.

2-9 Quand un grand nombre décimal doit être converti en binaire, il est parfois plus simple de le convertir en octal, puis de transformer en binaire ce dernier nombre. Faites l'essai avec 2313_{10} et comparez avec la méthode utilisée au problème 2-2 e).

SECTION 2-4

2-10 Convertissez en décimal ces nombres hexadécimaux.
a) 92 b) 1A6 c) 37FD d) 2C0 e) 7FF

2-11 Convertissez en hexadécimal ces nombres décimaux.
a) 75 b) 314 c) 2048 d) 25619 e) 4095

2-12 Convertissez en hexadécimal les nombres binaires du problème 2-1.

2-13 Trouvez l'équivalent binaire des nombres hexadécimaux du problème 2-10.

2-14 Dans la plupart des micro-ordinateurs, les *adresses* des emplacements mémoires sont exprimées en hexadécimal. Ces adresses sont des nombres séquentiels qui identifient chacune des cases mémoires.
a) Un certain micro-ordinateur peut stocker des nombres de 8 bits dans chacune de ses cases mémoires. Si l'intervalle des adresses mémoires va de 0000_{16} à $FFFF_{16}$, dites combien cet ordinateur a de cases mémoires.
b) Un autre micro-ordinateur possède 4096 emplacements en mémoire. Donnez l'intervalle de ses adresses exprimées en hexadécimal.

2-15 Donnez la suite des nombres entre 280 et 2A0.

SECTION 2-5

2-16 Codez en DCB les nombres décimaux suivants.
a) 47 b) 962 c) 187 d) 42,689,627 e) 1204

2-17 Donnez le nombre de bits nécessaire pour représenter les nombres décimaux de l'intervalle 0 à 999 selon le code binaire pur. Selon le code DCB.

2-18 Les nombres suivants sont des nombres DCB. Trouvez leur équivalent décimal.
a) 1001011101010010 c) 0111011101110101
b) 000110000100 d) 010010010010

SECTION 2-8

2-19 Exprimez en ASCII l'instruction « X=25/Y » (sans les guillemets).

2-20 Associez un bit de parité *paire* aux codes ASCII du problème 2-19 et donnez les résultats en hexadécimal.
D8, BD, B2, 35, AF, 59

2-21 Les groupes codés suivants doivent être transmis. Associez à chacun un bit de parité *paire*.
a) 10110110 b) 00101000 c) 11110111

SECTION 2-9

2-22 Convertissez les nombres décimaux que voici en leur équivalent DCB, puis associez à ces derniers un bit de parité *impaire*.
a) 74 b) 38 c) 165 d) 9201

2-23 Dans certains systèmes numériques, les nombres décimaux entre 000 et 999 sont exprimés suivant le code DCB. Ensuite un bit de parité *impaire* est ajouté à la fin de chaque groupe. Examinez les codes ci-dessous et supposez qu'ils viennent d'être transmis d'un endroit à un autre. Certains de ces groupes sont erronés. En prenant pour hypothèse qu'il n'y a jamais *plus de deux erreurs* dans ces groupes, trouvez les groupes où est survenue une seule erreur et ceux pour lesquels il y a sûrement eu deux erreurs. (*Suggestion*: Souvenez-vous que ce sont des nombres DCB).
a) 1001010110000
↑
└— bit de parité
b) 0100011101100
c) 0111110000011
d) 1000011000101

2-24 Supposons que le récepteur de l'exemple 2-10 reçoive les données suivantes:

0	1	0	0	1	0	0	0
1	1	0	0	0	1	0	1
1	1	0	0	1	1	0	0
1	1	0	0	1	0	0	0
1	1	0	0	1	1	0	0

Quelles erreurs le récepteur reconnaît-il dans ces données?

CONTRÔLE

2-25 Réalisez chacune des conversions suivantes. Pour certaines d'entre elles, vous procéderez peut-être de plusieurs façons pour savoir quelle méthode vous convient le mieux. Par exemple, certaines conversions binaire-décimal seront plus rapides directement, alors que d'autres se feront mieux si vous faites d'abord une conversion binaire-octal et ensuite une conversion octal-décimal.
a) 1417_{10} = ___ 2
b) 255_{10} = ___ 2
c) 11010001_2 = ___ 10
d) 1110101000100111_2 = ___ 10
e) 2497_{10} = ___ 8
f) 511_{10} = ___ 8
g) 235_8 = ___ 10
h) 4316_8 = ___ 10
i) $7A9_{16}$ = ___ 10
j) $3E1C_{16}$ = ___ 10
k) 1600_{10} = ___ 16
l) $38\ 187_{10}$ = ___ 16
m) 865_{10} = ___ (DCB)
n) 100101000111 (DCB) = ___ 10
o) 465_8 = ___ 16

p) $B34_{16}$ = ___ 8
q) 01110100 (DCB) = ___ 2
r) 111010_2 = ___ (DCB)

2-26 Exprimez le nombre décimal 37 dans les différents codes que voici:
a) code binaire pur, b) code DCB, c) code majoré de 3, d) ASCII (chaque chiffre considéré comme un caractère), e) nombre octal, f) nombre hexadécimal.

2-27 Complétez les phrases suivantes:
a) La conversion décimale- octal exige la division à répétition par 8.
b) La conversion décimale-hexadécimale exige la division à répétition par 16.
c) Dans le code DCB, chaque _____ est converti en son équivalent binaire de 4 bits.
d) Le code _____ a la caractéristique d'avoir seulement un bit qui change d'une valeur à une autre.
e) Un émetteur attache un _____ à un groupe codé afin de permettre au récepteur de détecter _____ .
f) Le code _____ est le code alphanumérique le plus courant dans les systèmes informatiques.
g) Les _____ et _____ sont souvent des façons commodes de représenter de grands nombres binaires.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 2-1

1. 2267 **2.** 32768

SECTION 2-2

1. 1010011 **2.** 1011011001

SECTION 2-3

1. 396 **2.** 222; 010010010 **3.** 235
4. 627, 630, 631 **5.** 1111001111 **6.** 699

SECTION 2-4

1. 9422 **2.** C2D; 110000101101 **3.** 97B5
4. E9E, E9F, EA0, EA1 **5.** 757

SECTION 2-5

1. 10110010_2; 000101111000 (DCB) **2.** 32
3. Avantage – facile à convertir; inconvénient – le DCB a besoin de plus de bits.

SECTION 2-7

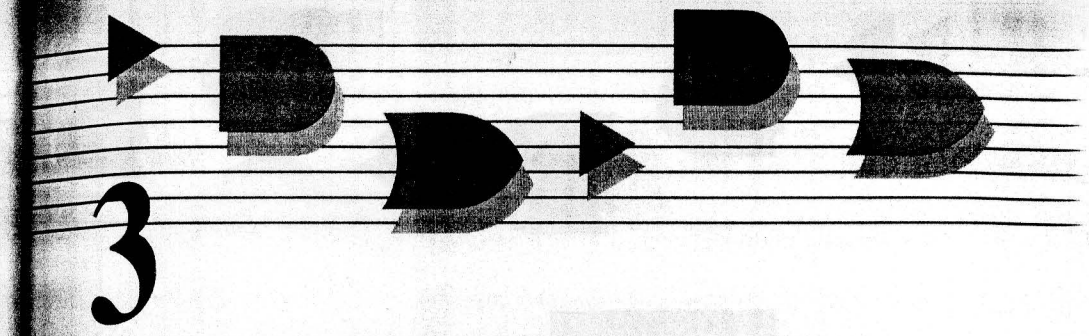
1. Pas d'états intermédiaires durant les transitions.

SECTION 2-8

1. 43, 4F, 55, 54, 20 3D, 20, 37, 32, 20, 24
2. STOP

SECTION 2-9

1. A4 **2.** 001101001 **3.** Deux erreurs dans la donnée ne modifient pas la valeur paire ou impaire du nombre de 1 dans la donnée.

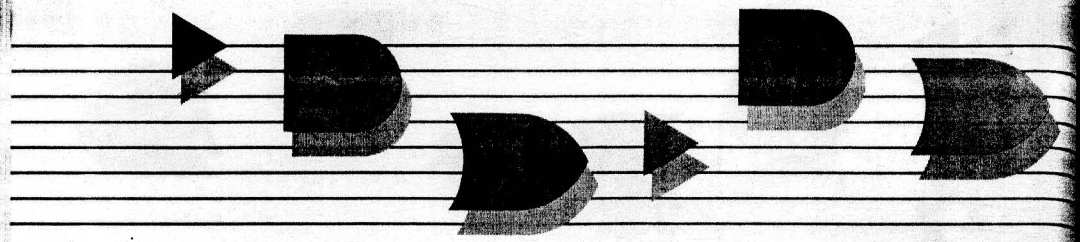


3

PORTES LOGIQUES ET ALGÈBRE BOOLÉENNE

Contenu

3-1 Constantes et variables booléennes
3-2 Tables de vérité
3-3 Opération OU
3-4 Opération ET
3-5 Opération NON
3-6 Mise sous forme algébrique des circuits logiques
3-7 Évaluation des sorties des circuits logiques
3-8 Matérialisation de circuits à partir d'expressions booléennes
3-9 Portes NI et portes NON-ET
3-10 Théorèmes de Boole
3-11 Théorèmes de De Morgan
3-12 Universalité des portes NON-ET et des portes NI
3-13 Représentations synonymes des portes logiques
3-14 Quelle représentation de porte doit-on choisir?
3-15 Symboles logiques normalisés IEEE/ANSI.



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Analyser un INVERSEUR.
- Décrire le fonctionnement des portes ET, NON-ET, OU et NI, et indiquer comment réaliser leur table de vérité.
- Tracer des chronogrammes des diverses portes de circuits logiques.
- Simplifier des circuits logiques complexes grâce à l'application des théorèmes et des identités de l'algèbre booléenne.
- Simplifier des équations booléennes confuses en recourant aux théorèmes de De Morgan.
- Recourir aux portes universelles (NON-ET ou NI) pour matérialiser le circuit exprimé par une expression booléenne.
- Exposer les avantages de construire un schéma de circuit logique avec les symboles synonymes plutôt qu'avec les symboles courants.
- Expliquer la notion de niveau vrai (BAS ou HAUT) d'un signal logique.
- Tracer et interpréter les nouvaux symboles normalisés IEEE/ANSI.

Introduction

Au Chapitre 1, nous avons mentionné que les circuits numériques fonctionnent en mode binaire, c'est-à-dire un mode dans lequel les tensions de sortie et d'entrée sont 0 ou 1; les valeurs 0 et 1 correspondent à des plages de tensions définies à l'avance. Voilà une caractéristique des circuits logiques qui nous permet de recourir à l'*algèbre de Boole* pour l'analyse et la conception de systèmes numériques. Dans ce chapitre, nous étudierons les *portes logiques*, qui constituent les blocs élémentaires des circuits logiques et nous verrons comment il est possible de décrire leur fonctionnement grâce à l'algèbre booléenne. Aussi, nous vous apprendrons comment on réussit à construire des circuits logiques en combinant des portes et comment l'algèbre de Boole parvient à décrire et à analyser ces derniers.

3-1 CONSTANTES ET VARIABLES BOOLÉENNES

L'algèbre booléenne se distingue principalement de l'algèbre ordinaire par des constantes et des variables qui ne peuvent prendre que les deux valeurs possibles 0 et 1. Une variable booléenne est une grandeur qui peut, à des moments différents, avoir la valeur 1 ou 0. Les variables booléennes servent souvent à représenter un niveau de tension sur un fil ou aux bornes d'entrée ou de sortie d'un circuit. Par exemple, dans un certain circuit numérique, on pourra avoir attribué la valeur booléenne 0 à l'intervalle de tensions 0 à 0,8 V, et la valeur booléenne 1 à l'intervalle 2 à 5 V*.

Ainsi, les valeurs booléennes 0 et 1 ne représentent pas des nombres réels mais plutôt l'état d'une variable électrique ou ce qui est convenu d'appeler un **niveau logique**. On dit que la tension d'un circuit numérique est au niveau logique 1 ou au niveau logique 0, selon la valeur réelle de cette tension. Dans le domaine de la logique numérique, on utilise d'autres expressions qui sont synonymes de 0 et 1. Certaines de ces expressions sont représentées au tableau 3-1. Pour notre part, nous utiliserons surtout 0/1 et NIVEAU HAUT/NIVEAU BAS.

TABLEAU 3-1

Niv. logique 0	Niv. logique 1
Faux	Vrai
Arrêt	Marche
Bas	Haut
Non	Oui
Ouvert	Fermé

L'algèbre de Boole est un outil qui permet d'exprimer les effets qu'ont les divers circuits numériques sur les entrées logiques et de manipuler les variables logiques en vue de déterminer la meilleure façon de matérialiser une certaine fonction logique. Dans le reste de ce livre, nous allons utiliser des lettres comme symboles pour représenter les variables logiques. Par exemple, A pourra correspondre à une entrée ou à une sortie d'un certain circuit numérique: en tout temps, A sera égal soit à 1 ou à 0.

Parce qu'il n'y a que deux valeurs possibles, l'algèbre booléenne se manipule plus aisément que l'algèbre ordinaire. En algèbre booléenne, il n'y a pas de fraction, de partie décimale, de nombre négatif, de racine carrée, de racine cubique, de logarithmes, de nombre imaginaire... En fait, dans cette algèbre, on ne retrouve que trois opérations élémentaires:

1. L'*addition logique*, dite aussi *opération OU*. Le symbole habituel de cette opération est le signe (+).
2. La *multiplication logique*, dite aussi *opération ET*. Son symbole habituel est le signe de la multiplication (·).
3. La *complémentation ou l'inversion logique*, dite aussi *opération NON*. Son symbole habituel est une barre de surlignement (−).

3-2 TABLES DE VÉRITÉ

De nombreux circuits logiques possèdent plusieurs entrées mais seulement une sortie. Une *table de vérité* nous fait connaître la réaction d'un circuit logique (sa valeur de sortie) aux diverses combinaisons de niveaux logiques appliqués aux entrées. La figure 3-1 nous montre des tables de vérité à deux, trois et quatre colonnes d'entrées.

* Les tensions comprises entre 0.8 et 2 V sont indéterminées (ne correspondent ni à 1 ni à 0) et ne doivent jamais survenir.

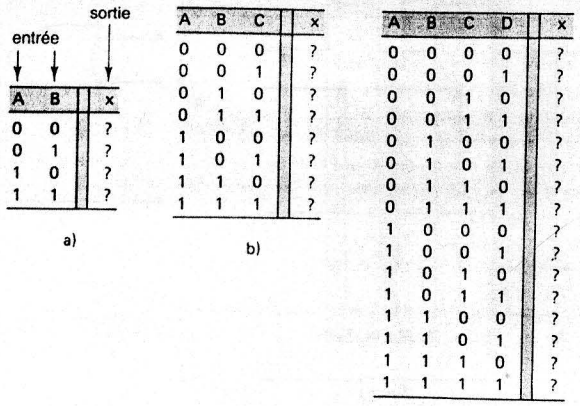


FIGURE 3-1 Tables de vérité: a) table à deux entrées; b) table à trois entrées; c) table à quatre entrées.

Dans chacune de ces tables, toutes les combinaisons possibles de 0 et de 1 pour les entrées (A, B, C, D) apparaissent à gauche, tandis que le niveau logique résultant de la sortie, x, est donné à droite. Pour le moment, il n'y a que des « ? » dans ces colonnes, car les valeurs de sortie sont différentes pour chaque type de circuit.

Notez que dans la table de vérité à deux entrées il y a quatre lignes, que dans celle à trois entrées il y a huit lignes et que dans la table à quatre entrées, il y en a seize. En général, pour une table de N entrées, il y a 2^N lignes. De plus, vous remarquez sans doute que la succession des combinaisons correspond à la suite du comptage binaire, de sorte que la détermination de toutes les combinaisons est directe et qu'on ne peut pas en oublier.

3-3 L'OPÉRATION OU

Soit deux variables logiques indépendantes, A et B. Quand on combine A et B au moyen de l'addition logique, le résultat x est exprimé par:

$$x = A + B$$

Dans cette équation, le signe + n'indique pas une addition ordinaire, mais plutôt l'addition logique (opération OU) dont les règles sont définies dans la table de vérité de la figure 3-2 a).

Vous déduisez peut-être de cette table que l'addition logique est identique à l'addition courante sauf pour le cas où $A = B = 1$. Car dans ce cas, la réponse de l'opération est 1 plutôt que 2 (comme dans l'addition ordinaire). C'est un résultat facile à retenir puisqu'en algèbre de Boole seules existent les valeurs 1 et 0, et que la plus grande valeur possible est 1. D'ailleurs, on obtiendra le même résultat pour l'équation $x = A + B + C$ quand $A = B = C = 1$. Donc

$$x = 1 + 1 + 1 = 1$$

Nous pouvons donc conclure que l'opération OU donne la valeur 1 quand l'une quelconque des variables d'entrée est 1. C'est une conclusion qui est évidente d'après la table de la figure 3-2 a).

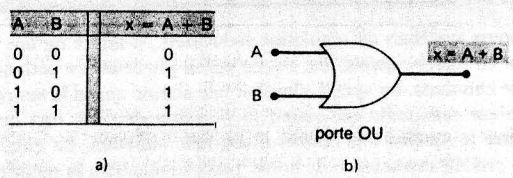


FIGURE 3-2 a) Table de vérité définissant l'opération OU; b) symbole de circuit servant à représenter une porte OU à deux entrées.

L'équation $x = A + B$ peut se traduire ainsi en langage clair: « x égale A plus B » ou « x égale A *OU* B ». Ces deux façons de traduire cette équation sont utilisées indistinctement. Ce qu'il faut surtout retenir c'est que le signe + signifie non pas une addition ordinaire mais une addition logique, définie par la table de vérité de la figure 3-2 a).

La porte OU En électronique numérique une *porte OU* est un circuit ayant au moins deux entrées et dont la sortie est égale à la somme logique (OU) des entrées. La figure 3-2 b) nous fait voir le symbole utilisé pour représenter une porte OU à deux entrées. Les entrées A et B sont des niveaux de tension logiques, et la sortie x est également un niveau de tension logique qui résulte de l'addition logique de A et B, d'où $x = A + B$. Autrement dit, la porte OU a un fonctionnement tel que sa sortie est à un niveau haut (niveau 1) quand *au moins* une de ses entrées est au niveau haut. À plus forte raison, quand les *deux* entrées sont au niveau haut. Cette porte sera au niveau bas (niveau 0) si toutes ses entrées sont au niveau bas.

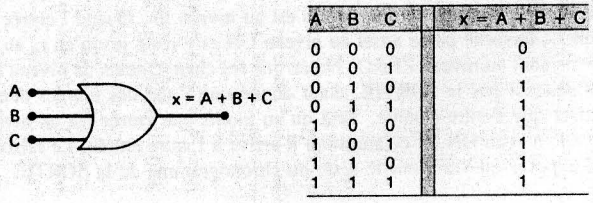
La même discussion s'applique intégralement à plus de deux entrées. La figure 3-3 montre une porte OU à trois entrées, ainsi que sa table de vérité. L'examen de cette table nous montre que la sortie sera 1 dans chaque cas où au moins une entrée est 1. Ce principe général est valable pour toutes les portes OU quel que soit le nombre d'entrées.

En utilisant le langage de l'algèbre de Boole, on peut exprimer la sortie x sous la forme $x = A + B + C$, où les signes + signifient toujours une addition logique. La sortie d'une porte OU quelconque a donc comme expression le résultat de l'opération OU appliquée à ses entrées. C'est un résultat que nous saurons exploiter plus tard au moment de l'analyse des circuits logiques.

Récapitulation de l'opération OU Voici les points importants à retenir au sujet de l'opération OU et de la porte OU:

1. L'opération OU donne un 1 *si l'une* quelconque de ses variables d'entrée est 1.
2. L'opération OU donne un 0 *si toutes* ses variables d'entrée sont 0.
3. Dans l'opération OU, $1 + 1 = 1$, $1 + 1 + 1 = 1$, etc.

FIGURE 3-3 Symbole et table de vérité d'une porte OU à trois entrées.



EXEMPLE 3-1

Dans de nombreux systèmes de régulation industriels, il arrive qu'une fonction de sortie doive se mettre en marche quand une entrée parmi plusieurs est actionnée. Par exemple, dans un procédé chimique, on veut déclencher une alarme quand la température du procédé excède une valeur maximale *ou* quand la pression dépasse une certaine limite. La figure 3-4 montre le schéma fonctionnel d'une telle situation. Le circuit du transducteur de température produit une tension de sortie proportionnelle à la température du procédé. Cette tension, V_T, est comparée à la tension de la température de consigne, V_{TR}, au moyen d'un circuit comparateur de tensions. La sortie de ce dernier est normalement une tension faible (niveau logique 0), mais qui devient haute (niveau logique 1) chaque fois que V_T excède V_{TR}, ce qui est une indication que la température du procédé est excessive. Un montage similaire sert à surveiller la pression: le comparateur de ce montage voit sa sortie passer d'un niveau bas à un niveau haut quand la pression devient excessive.

Pour qu'une alerte soit lancée si la température *ou* la pression dépasse le niveau acceptable, on peut appliquer les entrées des deux comparateurs à une porte OU. La sortie de cette porte passera au niveau haut (1) au moment de l'une ou de l'autre des conditions d'alerte et déclenchera l'alarme. La même solution peut évidemment être appliquée à des situations où interviennent plus de deux variables de procédé.

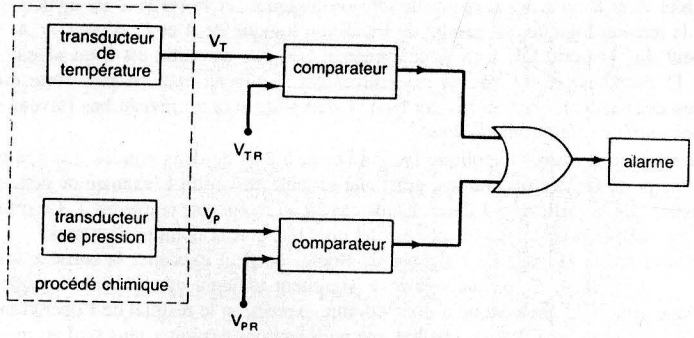


FIGURE 3-4 Exemple d'application d'une porte OU à un système d'alarme.

EXEMPLE 3-2

Trouvez la sortie de la porte OU de la figure 3-5. Les entrées de cette porte OU sont A et B et elles varient comme les chronogrammes illustrés. Par exemple, A est à ZÉRO en t_0 passe à UN en t_1, revient à ZÉRO en t_3, et ainsi de suite.

Solution On détermine la sortie de la porte OU en réalisant qu'elle est au niveau UN chaque fois que l'*une* ou l'*autre* de ses entrées est au niveau un. Quand l'entrée A passe au niveau UN en t_1, la sortie passe aussi au niveau UN et y reste jusqu'en t_4 au moment ou les deux entrées sont au niveau ZÉRO. Notez que les changements de niveau survenus en t_2 et t_3 n'influencent pas la SORTIE, étant donné que l'une des entrées demeure au niveau UN pendant que l'autre change. Tant qu'au moins une entrée est au niveau UN, la sortie conserve le niveau UN, et cela quoiqu'il arrive à l'autre entrée. Le raisonnement que l'on vient d'exposer est vrai pour le reste du chronogramme de la SORTIE.

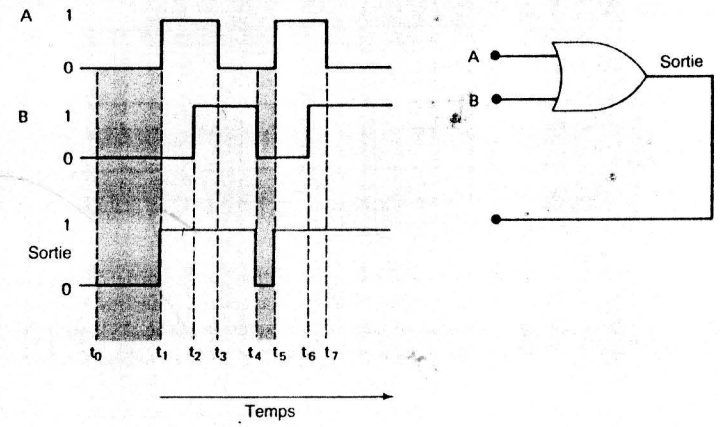


FIGURE 3-5 Exemple 3-2.

EXEMPLE 3-3A

Pour la situation illustrée à la figure 3-6, déterminez la forme d'onde de la sortie de la porte OU.

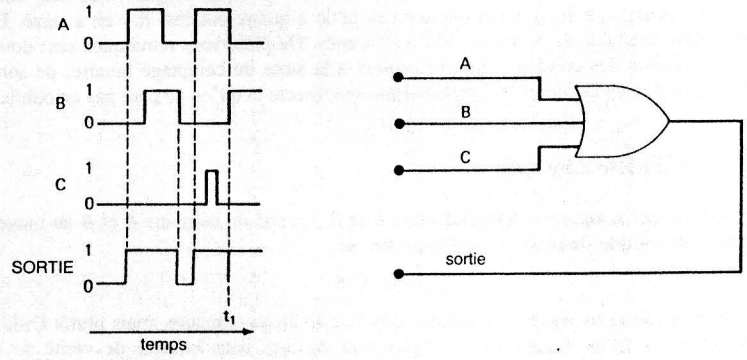


FIGURE 3-6

Solution Les trois entrées *A*, *B*, *C* de la porte OU changent comme les diagrammes des formes d'ondes. La sortie de cette porte est au niveau haut à chaque fois qu'une des entrées est au niveau haut. Sachant cela, on peut sans difficulté se convaincre que cette sortie varie comme le diagramme « SORTIE » de la figure. Remarquez particulièrement ce qui se passe au temps t_1. Sur le diagramme, on voit qu'à cet instant l'entrée *A* passe de 1 à 0 et que l'entrée *B* passe de 0 à 1. Comme ces entrées permutent approximativement au même moment et que ces transitions prennent un certain temps, pendant un court instant les deux entrées de la porte OU sont dans un état indéterminé compris entre 0 et 1. Cet état

indéterminé se répercute sur la sortie de la porte OU , comme l'indique le décrochement ou la pointe de la forme d'onde de sortie au moment t_1. Cette pointe et sa taille (amplitude et largeur) dépendent de la rapidité de la transition.

EXEMPLE 3-3B

Qu'arrive-t-il au décrochement observé sur la sortie si l'entrée C se stabilise au niveau HAUT pendant que A et B subissent des changements à l'instant t_1?

Solution Si l'entrée C est au niveau HAUT à l'instant t_1, la sortie de la porte ou conserve le niveau HAUT et cela quels que soient les changements dans les autres entrées, pour la simple raison qu'une entrée au niveau HAUT garde la sortie de la porte OU au niveau HAUT. Par conséquent, le décrochement ou pointe de tension n'est plus observé.

QUESTIONS DE RÉVISION

1. Dites quel est le seul ensemble de conditions qui produit un niveau BAS pour toute porte OU.
2. Donnez l'expression booléenne d'une porte OU à six entrées.
3. Si l'entrée *A* de la figure 3-6 est continuellement à 1, tracez la forme d'onde de sortie résultante.

3-4 L'OPÉRATION ET

Si deux variables logiques *A* et *B* sont combinées par la multiplication logique (opération ET), le résultat *x* s'exprime symboliquement ainsi:

$$x = A \cdot B$$

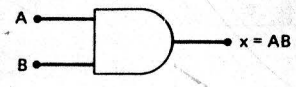
Dans cette expression, le signe · signifie l'opération booléenne ET, dont les règles d'opération sont données dans la table de vérité de la figure 3-7 a).

D'après cette table, vous devriez pouvoir déduire facilement que la multiplication logique est *exactement* comme la multiplication ordinaire. Quand *A* ou *B* est 0, le produit est nul; quand *A et B* sont 1, leur produit est 1. Il nous est donc possible d'affirmer que dans l'opération ET la réponse est 1 *si et seulement si* toutes les entrées sont 1, et qu'elle est 0 dans tous les autres cas.

FIGURE 3-7 a) Table de vérité de l'opération ET; b) symbole d'une porte ET.

A	B	x = A · B
0	0	0
0	1	0
1	0	0
1	1	1

a)



b)

A	B	C	x = ABC
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

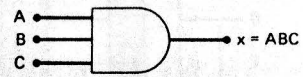


FIGURE 3-8 Table de vérité et symbole d'une porte ET à trois entrées.

L'expression $x = A \cdot B$, qui se lit « *x* égale *A et B* », est souvent abrégée en $x = AB$, comme en algèbre ordinaire où le signe multiplication est généralement omis. Le principal élément à ne pas oublier est le fait que l'opération ET est identique à la multiplication courante, mais que les variables ne peuvent prendre que les valeurs 0 et 1.

La porte ET La figure 3-7 b) nous fait voir une porte ET à deux entrées. La sortie de cette porte est égale au produit logique (ET) des entrées logiques, c'est-à-dire $x = AB$. Exprimée autrement, la porte ET est un circuit qui fonctionne pour que sa sortie soit à 1 seulement quand toutes ses entrées sont aussi à 1. Dans tous les autres cas, la sortie de la porte ET est à 0.

L'opération des portes ET à plus de deux entrées est analogue à ce qu'on vient de dire. Par exemple, on peut voir à la figure 3-8 une opération ET à trois entrées avec en vis-à-vis sa table de vérité. Comme précédemment, vous remarquez que la sortie est 1 seulement si $A = B = C = 1$. L'expression mathématique de cette sortie est $x = ABC$; pour une opération ET à quatre entrées, elle devient $x = ABCD$; etc.

Notez la différence entre le symbole de la porte ET et le symbole de la porte OU. Chaque fois que vous verrez le symbole de la porte ET dans un schéma de circuit logique, vous saurez que la sortie ne prend le niveau HAUT que *si toutes* les entrées sont au niveau HAUT. Par ailleurs, la présence du symbole de la porte OU vous indiquera que la sortie correspondante est au niveau HAUT quand *au moins une* des entrées est au niveau HAUT.

Récapitulation de l'opération ET

1. L'opération ET suit les mêmes règles que la multiplication ordinaire des 0 et des 1.
2. La sortie est égale à 1 dans le seul cas où toutes les entrées sont à 1.
3. La sortie est à 0 si au moins une entrée est à 0.

EXEMPLE 3-4

Déterminez la sortie *x* de la porte ET de la figure 3-9 pour les formes d'ondes illustrées.

Solution La sortie d'une porte ET est obtenue si on réalise qu'elle passe au niveau haut quand toutes ses entrées se trouvent simultanément au niveau haut. Pour les formes d'ondes indiquées, cette condition ne survient que durant les intervalles t_2-t_3 et t_6-t_7. À tous les autres instants, une ou plusieurs entrées est à 0, ce qui garde la sortie au niveau bas. Observez comment un changement dans une entrée pendant qu'une autre entrée reste à zéro n'a aucun effet sur la sortie.

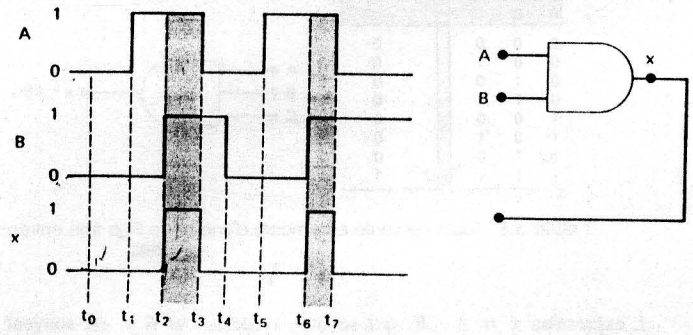


FIGURE 3-9 Exemple 3-4.

EXEMPLE 3-5A

Déterminez la forme d'onde de sortie de la porte ET illustrée à la figure 3-10.

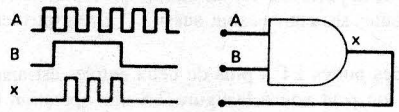


FIGURE 3-10 Exemple 3-5.

Solution La sortie x ne sera à 1 que si A et B sont tous les deux au niveau HAUT en même temps. Sachant cela, vous pouvez établir que la forme d'onde de x correspond à ce qui est illustré.

Vous remarquez que x est à 0 quand B est à 0, quelle que soit la valeur de A, et que x est analogue à A quand B vaut 1. De là, il est facile de s'imaginer que B est une entrée de *commande* qui détermine si l'onde A est transmise ou non à la sortie x. Utilisée de cette façon, la porte ET joue le rôle d'un circuit *de blocage*. On peut dire que $B = 0$ est la condition de blocage donnant une sortie de 0, et inversement que $B = 1$ est la condition de *validation* qui donne A en sortie. Cette fonction de blocage est une application importante des portes ET, sur laquelle nous reviendrons plus loin.

EXEMPLE 3-5B

Décrivez le comportement de l'onde de sortie x de la figure 3-10 si l'entrée B est maintenue au niveau 0?

Solution En gardant B au niveau BAS, la sortie x sera aussi gardée au niveau BAS. Il y a deux façons de démontrer cela: premièrement, avec $B = 0$, nous avons $x = A \cdot B = A \cdot 0 = 0$, puisque le fait de multiplier logiquement par 0 donne 0; deuxièmement, comme B est toujours à 0 et que l'on sait qu'il faut que toutes les entrées soient au niveau HAUT pour avoir une sortie au niveau HAUT, on conclut que la sortie est toujours au niveau BAS.

QUESTIONS DE RÉVISION

1. Indiquez la seule combinaison d'entrées d'une porte ET à cinq entrées qui produit un niveau HAUT en sortie.
2. Dites quel niveau logique doit être appliqué à la deuxième entrée d'une porte ET à deux entrées si on veut empêcher le signal logique de la première entrée d'atteindre la sortie.
3. *Vrai ou faux.* La sortie d'une porte ET diffère toujours de celle d'une porte OU pour les mêmes conditions d'entrée.

3-5 L'OPÉRATION NON

L'opération NON, contrairement aux opérations ET et OU, ne concerne qu'une variable d'entrée. Par exemple, si la variable A est soumise à une opération NON, le résultat x est donné par l'expression:

$$x = \overline{A}$$

où le trait de surlignement représente l'opération NON. Cette expression se lit « x égale NON A » ou « x égale l'*inverse* de A ou « x est le *complément* de A ». Ces trois descriptions sont d'usage courant et chacune signifie que la valeur logique de x, (soit $\overline{A}$), vaut l'inverse de la valeur logique de A. La table de vérité de la figure 3-11 a) décrit clairement ceci pour les deux cas $A = 0$ et $A = 1$. Donc:

$$\overline{1} = 0, \text{ car NON 1 est 0}$$

et

$$\overline{0} = 1, \text{ car NON 0 est 1.}$$

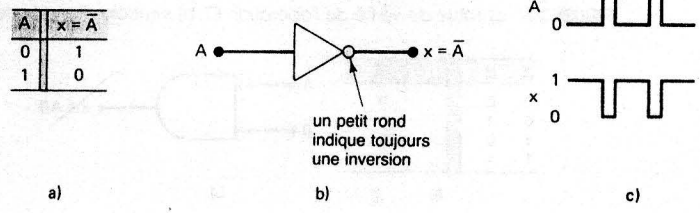
L'opération NON porte également le nom d'**inversion** ou de **complémentation**, noms que nous emploierons indifféremment tout au long de ce livre. Bien que nous utiliserons toujours le trait de surlignement, il est bon que vous sachiez qu'on trouve un autre signe pour indiquer une inversion: il s'agit de l'apostrophe ('). Donc

$$A' = \overline{A}$$

Ces deux façons de modifier la variable indiquent une fonction inversion.

Le circuit NON (INVERSEUR) On peut voir à la figure 3-11 b) le symbole d'un circuit NON, appelé plus couramment INVERSEUR. Un tel circuit n'a *toujours* qu'une entrée, et sa sortie prend le niveau logique opposé du niveau logique de l'entrée. Par ailleurs, la figure 3-11 c) nous fait voir comment un INVERSEUR change le signal d'entrée: chaque point de l'onde est inversé.

FIGURE 3-11 a) La table de vérité et b) le symbole du circuit NON, c) les formes d'ondes correspondantes.



Récapitulation des opérations booléennes Voici un résumé des règles qui régissent les opérations OU, ET et NON.

OU	*ET*	*NON*
$0 + 0 = 0$	$0 \cdot 0 = 0$	$\overline{0} = 1$
$0 + 1 = 1$	$0 \cdot 1 = 0$	$\overline{1} = 0$
$1 + 0 = 1$	$1 \cdot 0 = 0$	
$1 + 1 = 1$	$1 \cdot 1 = 1$	

QUESTIONS DE RÉVISION

1. La sortie de l'INVERSEUR de la figure 3-11 est raccordée à l'entrée d'un second INVERSEUR. Déterminez pour tous les niveaux de *A* le niveau de sortie du second INVERSEUR.
2. La sortie de la porte ET de la figure 3-7 est raccordée à l'entrée d'un INVERSEUR. Construisez la table de vérité de la sortie de l'INVERSEUR, *y*, pour toutes les combinaisons de *A* et *B*.

3-6 MISE SOUS FORME ALGÉBRIQUE DES CIRCUITS LOGIQUES

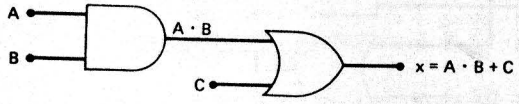
Tout circuit logique, quelle que soit sa complexité, peut être décrit au moyen des opérations booléennes déjà décrites parce que la porte ET, la porte OU et la porte NON sont les circuits constitutifs élémentaires des systèmes numériques. À titre d'exemple, considérons le circuit de la figure 3-12 comprenant trois entrées A, B et C et une seule sortie x. En recourant à l'expression booléenne de chacune des portes, on peut facilement trouver l'équation correspondant à la sortie.

La sortie de la porte ET a pour expression $A \cdot B$; cette combinaison est une entrée de la porte OU dont l'autre entrée est le signal C. Cette dernière porte a pour effet d'additionner logiquement ses entrées , ce qui donne comme expression de sortie $x = A \cdot B + C$. (Cette dernière équation aurait aussi bien pu s'écrire $x = C + A \cdot B$, puisque l'ordre des termes dans une fonction OU n'a pas d'importance.)

Parfois, il arrive qu'il y ait confusion sur l'opération d'une expression qui doit être effectuée en premier. Ainsi, l'équation $A \cdot B + C$ peut être interprétée de deux façons: 1) ou bien $A \cdot B$ est additionné logiquement (OU) avec C, 2) ou bien A et multiplié logiquement (ET) avec $B + C$. Pour lever cette indétermination, il est convenu que dans une expression contenant des opérateurs ET et OU, ce sont les opérateurs ET qui sont appliqués en premier, sauf s'il y a des *parenthèses*; dans ce cas, il faut évaluer avant toute chose l'expression entre parenthèses. Cette règle déterminant l'ordre des opérations est la même que celle en vigueur dans l'algèbre courante.

Comme exemple supplémentaire, considérons le circuit de la figure 3-13. Le résultat de la porte OU est simplement $A + B$. La sortie de cette porte aboutit à l'entrée de la porte ET, alors que l'autre entrée de cette dernière reçoit le signal C. L'expression de la sortie de la porte ET est donc $x = (A + B) \cdot C$. Notez l'emploi des parenthèses pour

FIGURE 3-12 Un circuit et son équation booléenne.



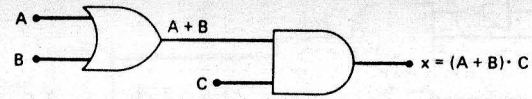


FIGURE 3-13 Un circuit logique dont l'expression de sortie comporte des parenthèses.

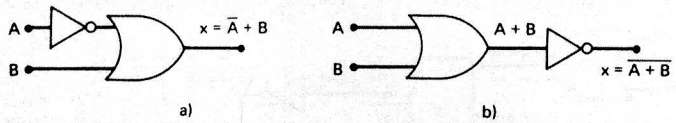


FIGURE 3-14 Circuits comprenant des INVERSEURS.

indiquer que A et B sont *d'abord* additionnés logiquement avant que leur somme OU soit multipliée logiquement avec C. Sans les parenthèses, notre interprétation serait erronée, car $x = A + B \cdot C$ signifie que A est réuni dans une porte OU avec le produit $B \cdot C$.

Circuits renfermant des INVERSEURS Chaque fois qu'un INVERSEUR se trouve dans le schéma d'un circuit logique, son équation est simplement l'expresssion de son entrée surmontée d'un trait. On trouve à la figure 3-14 deux exemples de circuits avec INVERSEURS. Dans la figure 3-14 a), l'entrée A passe par un INVERSEUR dont la sortie est $\overline{A}$. Cette sortie de l'INVERSEUR est réunie dans une porte OU avec B, de sorte que l'équation de sortie de cette porte est égale à $\overline{A} + B$. Remarquez que le trait est seulement au-dessus de A ce qui signifie que A est d'abord inversé puis additionné logiquement avec B.

Dans la figure 3-14 b), la sortie de la porte OU égale $A + B$, sortie qui vient alimenter un INVERSEUR. La sortie de ce dernier est donc $\overline{(A + B)}$, puisque *toute* l'expression est inversée. Notez que le trait de surlignement couvre toute l'expression $(A + B)$. C'est un élément important à remarquer, car nous verrons plus loin que $\overline{(A + B)}$ et $(\overline{A} + \overline{B})$ ne sont *pas* équivalents. En effet, $\overline{(A + B)}$ signifie que A et B sont réunis dans une porte OU, puis que leur somme est inversée, tandis que l'expression, $(\overline{A} + \overline{B})$ indique que A est inversé, que B est inversé et que ces deux résultats sont additionnés logiquement.

La figure 3-15 nous présente deux autres exemples, que vous devez étudier attentivement. Remarquez particulièrement les *deux* ensembles de parenthèses séparés de la figure 3-15 b). De plus, prenez note, dans la figure 3-15 a), du fait que la variable d'entrée A agit comme entrée dans deux portes différentes.

QUESTION DE RÉVISION

1. À la figure 3-15 a), remplacez la porte OU par une porte ET et les portes ET par des portes OU. Écrivez ensuite la nouvelle expression de sortie.

3-7 ÉVALUATION DES SORTIES DES CIRCUITS LOGIQUES

Quand on connaît l'expression booléenne de la sortie d'un circuit, il est possible de trouver le niveau logique correspondant à n'importe lesquelles des valeurs se trouvant aux entrées

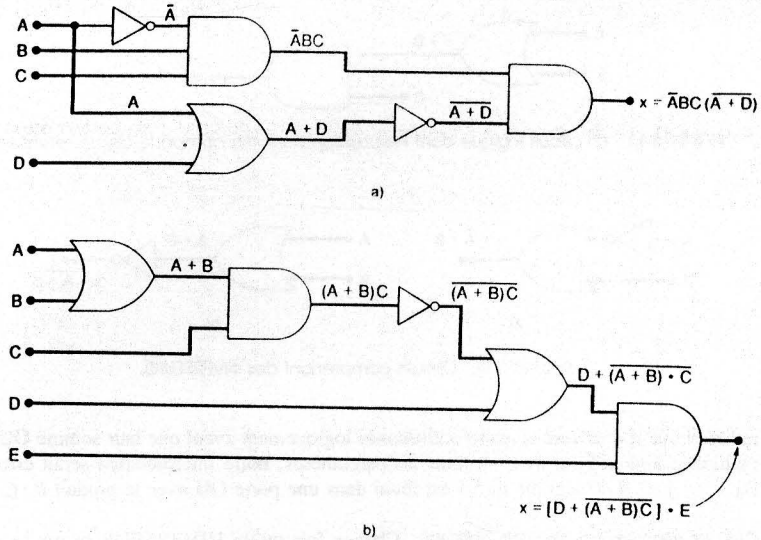


FIGURE 3-15 Exemples.

du circuit. Par exemple, supposons que l'on veuille connaître le niveau logique de la sortie x pour le circuit illustré à la figure 3-15 a) dans le cas où $A = 0$, $B = 1$, $C = 1$ et $D = 1$. Comme en algèbre ordinaire, on détermine la valeur de x en substituant aux variables de l'expression les valeurs données et en effectuant les opérations comme ceci:

$$\begin{aligned} x &= \bar{A}BC(\overline{A + D}) \\ &= \bar{0} \cdot 1 \cdot 1 \cdot (\overline{0 + 1}) \\ &= 1 \cdot 1 \cdot 1 \cdot (\overline{0 + 1}) \\ &= 1 \cdot 1 \cdot 1 \cdot (\bar{1}) \\ &= 1 \cdot 1 \cdot 1 \cdot 0 \\ &= 0 \end{aligned}$$

À titre d'illustration supplémentaire, évaluons la sortie du circuit de la figure 3-15 b) pour le cas $A = 0$, $B = 0$, $C = 1$, $D = 1$ et $E = 1$.

$$\begin{aligned} x &= [D + \overline{(A + B)C}] \cdot E \\ &= [1 + \overline{(0 + 0) \cdot 1}] \cdot 1 \\ &= [1 + \overline{0 \cdot 1}] \cdot 1 \\ &= [1 + \bar{0}] \cdot 1 \\ &= [1 + 1] \cdot 1 \\ &= 1 \cdot 1 \\ &= 1 \end{aligned}$$

On peut proposer, comme règle générale, pour l'évaluation d'une expression booléenne la marche à suivre que voici:

1. Premièrement, effectuer toutes les inversions ne touchant qu'une seule variable, ainsi $\bar{0} = 1$ ou $\bar{1} = 0$.
2. Ensuite, faire toutes les opérations entre parenthèses.
3. Évaluer une opération ET avant une opération OU, à moins que des parenthèses signalent le contraire.
4. Si une expression est surmontée d'une barre, évaluer l'expression d'abord puis inverser le résultat.

Détermination d'un niveau de sortie d'après un diagramme Le niveau logique d'une sortie pour des entrées données se détermine également directement d'après le diagramme du circuit *sans* qu'il soit nécessaire de recourir à l'expression booléenne. C'est une technique qu'utilisent souvent les techniciens durant le dépannage ou l'essai d'un système logique, puisqu'elle leur apprend la valeur de sortie finale de même que la sortie que doit avoir chacune des portes intermédiaires. Comme exemple, voyons la figure 3-16 qui correspond à la figure 3-15 a) redessinée, mais cette fois-ci avec l'indication des valeurs des entrées, soit $A = 0$, $B = 1$, $C = 1$ et $D = 1$. La marche à suivre consiste à commencer aux entrées et à progresser à travers chaque INVERSEUR et chaque porte, tout en inscrivant leur sortie, jusqu'à la détermination de la sortie finale.

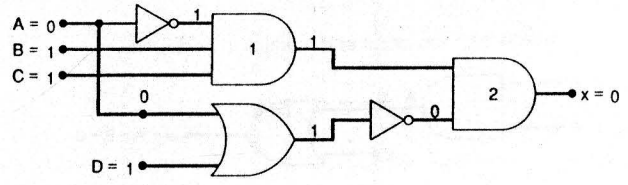
À la figure 3-16, la porte ET, appelée 1, voit un signal 1 sur ses *trois* entrées, parce que l'INVERSEUR a transformé le signal $A = 0$ en $\bar{A} = 1$. Cette condition d'entrée cause l'apparition d'un 1 à la sortie de la porte ET, car $1 \cdot 1 \cdot 1 = 1$. Les entrées de la porte OU sont 0 et 1 ce qui donne comme sortie la valeur 1, puisque $1 + 0 = 1$. Cette valeur est inversée et appliquée à la porte ET 2 en même temps que le 1 émis par la première porte ET. La combinaison d'un 1 et d'un 0 aux entrées de la porte ET 2 produit un 0, car $0 \cdot 1 = 0$.

EXEMPLE 3-6

Déterminez la sortie du circuit de la figure 3-16 pour le cas où toutes les entrées sont au niveau BAS.

Solution Quant A = B = C = D = 0, la sortie de la porte ET 1 est au niveau BAS. Ce niveau BAS place un niveau BAS sur l'entrée de la porte ET 2, ce qui automatiquement donne $x = 0$ quels que soient les niveaux des autres entrées du circuit. Cet exemple démontre qu'il n'est pas toujours obligatoire de trouver les niveaux logiques de tous les points pour connaître la sortie.

FIGURE 3-16 Détermination d'un niveau de sortie d'après un diagramme de circuit.



QUESTIONS DE RÉVISION

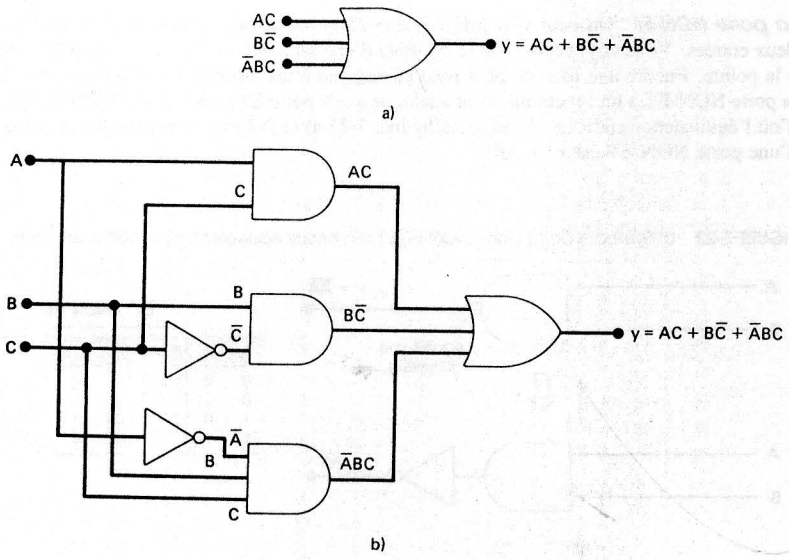
1. À l'aide de l'expression pour x, trouvez la valeur de sortie du circuit de la figure 3-15 a) pour les conditions $A = 0$, $B = 1$, $C = 1$ et $D = 0$.
2. À l'aide de l'expression pour x, trouvez la valeur de sortie du circuit de la figure 3-15 b) pour les conditions $A = B = E = 1$, $C = D = 0$.
3. Répondez aux questions 1 et 2 en établissant les niveaux logiques à chacune des entrées et des sorties des diverses portes, comme on l'a fait pour le circuit de la figure 3-16.

3-8 MATÉRIALISATION DE CIRCUITS À PARTIR D'EXPRESSIONS BOOLÉENNES

Si l'opération d'un circuit est définie par une expression booléenne, il est possible de tracer directement un diagramme logique à partir de cette expression. Par exemple, si on a besoin d'un circuit tel que $x = A \cdot B \cdot C$, on sait immédiatement qu'il nous faut une porte ET à trois entrées. Si nous voulons un circuit tel que $x = A + \overline{B}$, nous recourons à une porte OU à deux entrées dont l'une est transformée par un INVERSEUR. Le raisonnement qui nous a servi pour ces cas simples peut être étendu à des circuits plus complexes.

Supposons que l'on veuille construire un circuit dont la sortie est $y = AC + B\overline{C} + \overline{A}BC$. Cette expression booléenne est constituée de trois termes (AC, $B\overline{C}$, $\overline{A}BC$) qui sont additionnés logiquement. On déduit qu'il nous faut une porte OU à trois entrées auxquelles sont appliqués respectivement les signaux AC, $B\overline{C}$, $\overline{A}BC$. C'est d'ailleurs ce que nous fait voir la figure 3-17 a), qui représente une porte OU à trois entrées dont les entrées sont appelées AC, $B\overline{C}$ et $\overline{A}BC$.

FIGURE 3-17 Construction d'un circuit logique à partir d'une expression booléenne.



Chaque entrée de la porte OU est un produit logique, ce qui signifie qu'il a fallu trois portes ET alimentées par les entrées appropriées pour produire ces termes. C'est ce qu'on peut voir à la figure 3-17 b) où est tracé le schéma final du circuit. Notez l'emploi d'INVERSEURS pour produire les signaux $\overline{A}$ et $\overline{C}$ apparaissant dans l'expression.

Cette façon de procéder pourra toujours être suivie, bien que l'on vous enseignera, plus tard, des techniques plus directes et plus rapides d'arriver au même résultat. Pour le moment, c'est cette méthode systématique que nous utiliserons afin de ne pas vous montrer trop de choses à la fois.

EXEMPLE 3-7

Tracez le diagramme du circuit qui matérialise l'expression $x = AB + \overline{B}C$.

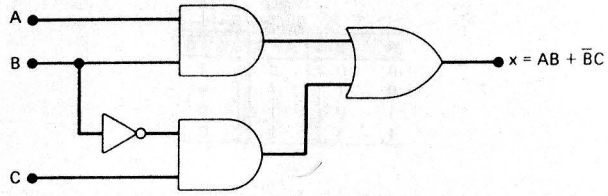


FIGURE 3-18 Exemple 3-7.

Solution Cette expression contient les termes AB et $\overline{B}C$ qui sont les entrées d'une porte OU; chacun de ces termes est à son tour le résultat d'une multiplication logique (par des portes ET). La réponse est le circuit de la figure 3-18.

QUESTION DE RÉVISION

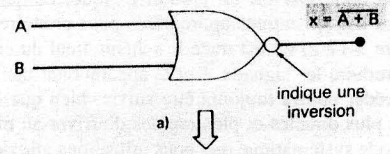
1. Tracez le diagramme du circuit qui matérialise l'expression $x = \overline{A}BC(\overline{A + D})$ en n'utilisant pas de portes ayant plus de trois entrées.

3-9 PORTES NI ET PORTES NON-ET

En technique numérique, on retrouve très souvent deux autres types de portes logiques: la porte NI et la porte NON-ET. En réalité, ces portes correspondent à des combinaisons d'opérations élémentaires ET, OU et NON, et il est relativement facile de les décrire au moyen des fonctions de l'algèbre booléenne que vous connaissez déjà.

La porte NI On peut voir à la figure 3-19 a) le symbole d'une porte NI à deux entrées. Vous constaterez que c'est le symbole d'une porte OU sauf qu'il y a un petit rond à la pointe. Ce petit rond correspond à une opération d'inversion. Ainsi, la porte NI a un fonctionnement analogue à une porte OU suivie d'un INVERSEUR, d'où l'équivalence entre les circuits des figures 3-19 a) et 3-19 b). L'expression de sortie d'une porte NI est $x = \overline{A + B}$.

La table de vérité montrée à la figure 3-19 c) nous apprend que la sortie d'une porte NI est exactement l'inverse de celle d'une porte OU pour toutes les conditions des entrées possibles. Une porte OU prend le niveau HAUT, si au moins une des ses entrées est au



		OU	NI
A	B	A + B	A + B
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

c)

FIGURE 3-19 a) Symbole de la porte NI; b) son circuit équivalent; c) sa table de vérité.

niveau HAUT, tandis que la porte NI prend le niveau BAS si au moins une de ses entrées est au niveau HAUT. Le même fonctionnement se retrouve dans les portes NI à plus de deux entrées.

EXEMPLE 3-8

Déterminez la forme d'onde de sortie d'une porte NI pour les formes d'ondes d'entrée de la figure 3-20.

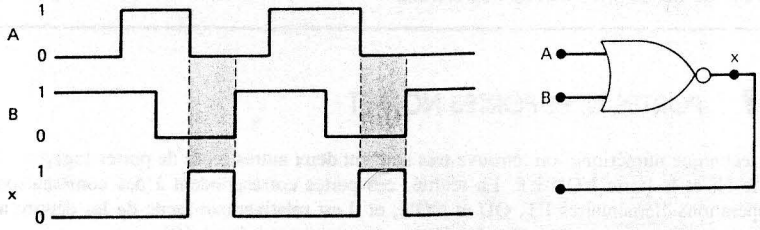


FIGURE 3-20 Exemple 3-8.

Solution Il y a plusieurs façons de déterminer la forme d'onde de sortie d'une porte NI. Une première façon consiste à trouver l'onde de sortie d'une porte OU puis de l'inverser (intervertir les 0 et les 1). Une autre façon consiste à exploiter le fait que la sortie d'une porte NI prend un niveau HAUT *seulement* si toutes ses entrées sont aux niveaux BAS. Ainsi, on peut examiner les ondes des entrées, trouver les périodes durant lesquelles toutes les entrées sont BASSES et faire correspondre un niveau HAUT à la sortie de la porte NI pour ces périodes. Cette dernière sortie sera basse pendant tout le reste du temps. La forme d'onde résultante est illustrée sur la figure.

EXEMPLE 3-9

Établissez l'expression booléenne pour une porte NI à trois entrées suivie d'un INVERSEUR.

Solution Reportez-vous à la figure 3-21 où se trouve le diagramme du circuit de l'exemple. L'équation de la sortie de la porte NI est $(\overline{A + B + C})$, dont le signal correspondant est appliqué à un INVERSEUR pour produire

$$x = \overline{(\overline{A + B + C})}$$

Le double signe d'inversion nous indique que la grandeur $(A + B + C)$ a été inversée une première fois, puis inversée une seconde fois. Il doit être évident pour vous que l'expression $(A + B + C)$ est demeurée inchangée. Donc

$$x = \overline{(\overline{A + B + C})} = A + B + C$$

Toutes les fois que la même variable est surmontée de deux barres, celles-ci s'annulent, comme dans l'exemple ci-dessus. Toutefois dans les cas comme $\overline{\overline{A} + \overline{B}}$, les barres de l'opérateur inversion ne s'annulent pas. Ceci du fait que les petites barres complémentent respectivement les variables A et B, alors que la grande barre complémente la grandeur $(\overline{A} + \overline{B})$. Donc $\overline{\overline{A} + \overline{B}} \neq A + B$; de même $\overline{\overline{AB}} \neq AB$.

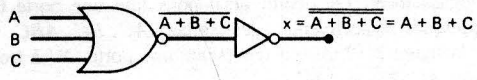
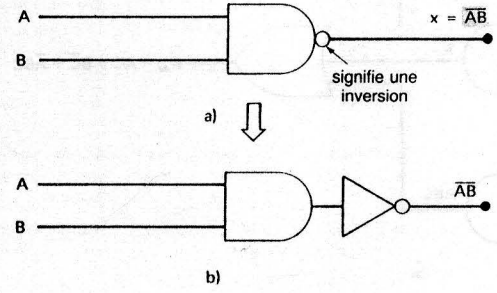


FIGURE 3-21 Exemple 3-9.

La porte NON-ET On peut voir à la figure 3-22 a) le symbole d'une porte NON-ET à deux entrées. Vous voyez que c'est le symbole d'une porte ET sauf qu'il y a un petit rond à la pointe. Encore une fois, ce petit rond correspond à une opération d'inversion. Ainsi, la porte NON-ET a un fonctionnement analogue à une porte ET suivie d'un INVERSEUR, d'où l'équivalence entre les circuits des figures 3-22 a) et 3-22 b). L'expression de sortie d'une porte NON-ET est $x = \overline{AB}$.

FIGURE 3-22 a) Symbole de la porte NON-ET; b) son circuit équivalent; c) sa table de vérité.



		ET	NON-ET
A	B	AB	AB
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

c)

La table de vérité de la figure 3-22 c) nous fait voir que la sortie d'une porte NON-ET est exactement l'inverse de celle de la porte ET pour toutes les conditions d'entrée possibles. La sortie de la porte ET prend le niveau HAUT quand toutes ses entrées sont au niveau HAUT, tandis que la porte NON-ET prend le niveau BAS quand toutes ses entrées sont au niveau HAUT. Et ceci reste vrai quel que soit le nombre d'entrées de la porte NON-ET.

EXEMPLE 3-10

Déterminez la forme d'onde de sortie d'une porte NON-ET pour les formes d'ondes des entrées de la figure 3-21.

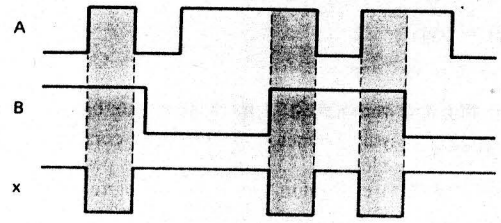


FIGURE 3-21 Exemple 3-10.

Solution Il y a plusieurs façons de résoudre ce problème. Une première consiste à tracer la sortie d'une porte ET, puis à inverser le tout. Une autre façon exploite le fait que la sortie d'une porte NON-ET prend le niveau BAS quand toutes ses entrées sont au niveau HAUT. Ainsi, il suffit de déterminer les intervalles où les entrées sont toutes à 1 et de mettre un 0 en sortie pour ces intervalles. Tout le reste du temps la sortie sera à 1.

EXEMPLE 3-11

Réalisez le circuit logique correspondant à l'expression $x = \overline{AB \cdot (\overline{C + D})}$ en n'utilisant que des portes NI et NON-ET.

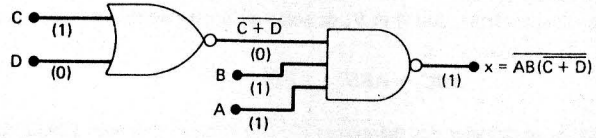


FIGURE 3-24 Exemple 3-11.

Solution Le terme $(\overline{C + D})$ est l'expression de la sortie d'une porte NI. Cette expression est multipliée logiquement par A et B, puis le résultat est inversé. Évidemment, cette dernière opération est celle d'une porte NON-ET. Donc le circuit recherché est celui de la figure 3-24. Notez que la porte NON-ET, dans un premier temps, multiplie logiquement les termes A, B et $(\overline{C + D})$, et, dans un second temps, elle inverse le résultat *total*.

EXEMPLE 3-12

Trouvez le niveau de sortie du circuit de la figure 3-24 pour les conditions $A = B = C = 1$ et $D = 0$.

Solution Une première méthode consiste à évaluer l'expression donnant x.

$$\begin{aligned} x &= \overline{AB(\overline{C + D})} \\ &= \overline{1 \cdot 1 \cdot (\overline{1 + 0})} \\ &= \overline{1 \cdot 1 \cdot (\overline{1})} \\ &= \overline{1 \cdot 1 \cdot 0} \\ &= \overline{0} = 1 \end{aligned}$$

Une seconde méthode consiste à écrire sur le schéma entre parenthèses les niveaux logiques des entrées et à suivre leur changement à travers chacune des portes jusqu'au résultat final. Les entrées de la porte NI sont 1 et 0, ce qui produit un 0 en sortie (car une porte OU, pour les mêmes conditions, donne 1). Donc les entrées de la porte NON-ET sont 0, 1 et 1, ce qui donne un résultat de 1 (une porte ET donne, pour ces conditions, une sortie de 0).

QUESTIONS DE RÉVISION

1. Quel est le seul ensemble de conditions d'entrées d'une porte NI à trois entrées qui donnera un niveau HAUT en sortie?
2. Dites quelle porte est équivalente à une porte NON-ET suivie d'un INVERSEUR?
3. Si vous remplacez la porte NI de la figure 3-24 par une porte NON-ET et la porte NON-ET par une porte NI, indiquez la nouvelle expression de x?

3-10 THÉORÈMES DE BOOLE

Nous avons vu comment l'algèbre de Boole peut servir à analyser un circuit logique et à exprimer ce dernier sous forme mathématique. Poursuivons maintenant notre étude de l'algèbre booléenne en examinant les théorèmes de Boole qui sont des règles qu'on utilise pour simplifier les expressions logiques et, par le fait même, les circuits logiques. La figure 3-25 présente le premier groupe de théorèmes dans lesquels x est une variable logique prenant soit la valeur 0, soit la valeur 1. Sur cette figure, il y a vis-à-vis de chaque théorème un schéma de circuit logique qui démontre le théorème.

Le théorème (1) affirme que la mise en fonction ET de toute variable avec 0 donne nécessairement 0. Cela est facile à démontrer, puisque l'on sait que l'opération ET est équivalente à la multiplication ordinaire dans laquelle la multiplication de n'importe quelle valeur par 0 donne 0. De plus, nous avons appris que la sortie d'une porte ET est 0 si au moins une de ses entrées est 0, malgré les valeurs présentes sur les autres entrées.

Le théorème (2) est évident si on utilise la comparaison avec la multiplication ordinaire.

Le théorème (3) peut être démontré en examinant toutes les possibilités: si $x = 0$, alors $0 \cdot 0 = 0$ et si $x = 1$, alors $1 \cdot 1 = 1$, d'où $x \cdot x = x$.

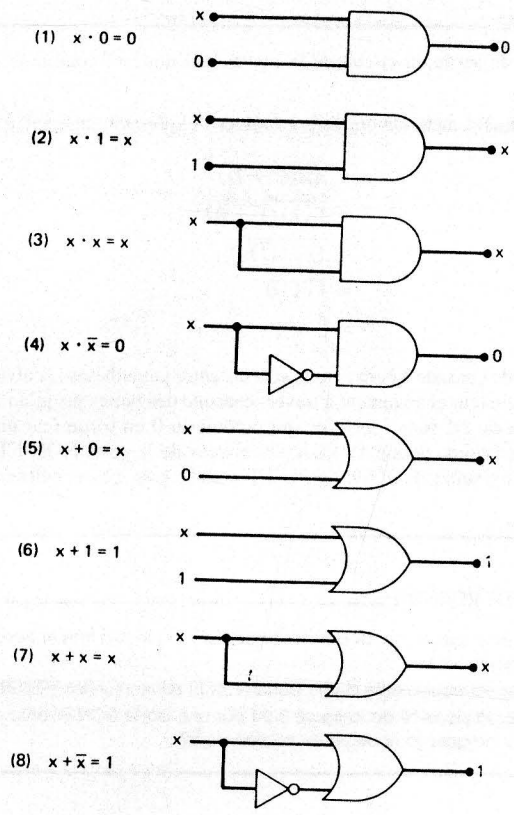


FIGURE 3-25 Théorèmes pour une variable.

La démonstration du théorème (4) est analogue à la précédente. Il est toutefois possible d'appliquer le raisonnement suivant: en tout temps, il faut que x ou bien son complément $\bar{x}$ soit égal à 0, de sorte que leur produit logique est toujours nul.

Le théorème (5) est immédiat puisque le fait d'*additionner* un 0 à n'importe quelle valeur ne change rien, aussi bien dans le cas de l'addition ordinaire que dans celui de l'addition logique.

Le théorème (6) établit que la mise en fonction OU de toute variable avec 1 donne toujours 1. On peut vérifier que cela est vrai pour les deux valeurs de x: $0 + 1 = 1$ et $1 + 1 = 1$. Voici une autre démonstration: si vous vous rappelez bien, la sortie d'une porte OU donne toujours 1 si *au moins une* de ses entrées est 1, et cela malgré les valeurs appliquées aux autres entrées.

Le théorème (7) se démontre sans difficulté en étudiant les deux possibilités de x: $0 + 0 = 0$ et $1 + 1 = 1$.

La preuve du théorème (8) est analogue à celle du théorème (7). Toutefois, une autre façon de le démontrer est d'appliquer le raisonnement suivant: en tout temps, il faut que x ou son complément $\bar{x}$ soit égal à 1, de sorte que dans ce théorème on met toujours en fonction OU un 0 et un 1, ce qui donne toujours 1.

Avant de vous présenter les autres théorèmes de Boole, nous tenons à mentionner que dans les théorèmes (1) à (8), la variable x peut correspondre à une expression renfermant plus d'une variable. Par exemple, si nous avons $A\bar{B}(\overline{A\bar{B}})$, nous pouvons affirmer, en posant $x = A\bar{B}$ et d'après le théorème (4), que $A\bar{B}(\overline{A\bar{B}}) = 0$.

Théorèmes pour plusieurs variables Les théorèmes suivants portent sur plus d'une variable:

(9) $x + y = y + x$

(10) $x \cdot y = y \cdot x$

(11) $x + (y + z) = (x + y) + z = x + y + z$

(12) $x(yz) = (xy)z = xyz$

(13a) $x(y + z) = xy + xz$

(13b) $(w + x)(y + z) = wy + xy + wz + xz$

(14) $x + xy = x$

(15) $x + \bar{x}y = x + y$

Les théorèmes (9) et (10) montrent que ET et OU sont des *lois de composition commutatives*, donc que l'ordre de la multiplication ou de l'addition logique de deux variables n'a pas d'importance, que le résultat reste le même.

Les théorèmes (11) et (12) montrent que ET et OU sont des *lois de composition associatives*, qui indiquent que l'on peut grouper, comme l'on veut, les variables dans une expression de multiplication ou d'addition logique.

Le théorème (13) fait voir que la multiplication logique est *distributive* par rapport à l'addition logique, c'est-à-dire que l'on peut développer une expression en la multipliant terme à terme, tout comme dans l'algèbre ordinaire. Ce théorème démontre également que l'on peut mettre en facteur une expression. On veut dire par là que si nous avons une somme de termes, chacun renfermant une variable commune, il est possible de mettre cette variable en facteur, comme on le fait en algèbre ordinaire. Par exemple, considérons l'expression $A\bar{B}C + \bar{A}\bar{B}\bar{C}$; il est possible de mettre en facteur $\bar{B}$:

$$A\bar{B}C + \bar{A}\bar{B}\bar{C} = \bar{B}(AC + \bar{A}\bar{C})$$

Voici un autre exemple: $ABC + ABD$. Dans cette expression, il y a deux variables communes dans chaque terme, soit A et B, de sorte qu'il est possible de mettre en facteur $A \cdot B$. Il vient:

$$ABC + ABD = AB(C + D)$$

Il est facile de se rappeler des théorèmes (9) à (13) puisqu'ils sont identiques à ceux de l'algèbre ordinaire. Par contre, les théorèmes (14) et (15) ne se retrouvent pas en algèbre ordinaire. On les démontre en passant en revue toutes les possibilités de x et de y. Voici la démonstration du théorème (14):

1er cas: Pour $x = 0$, $y = 0$,

$$x + xy = x$$

$$0 + 0 \cdot 0 = 0$$

$$0 = 0$$

2e cas: Pour $x = 0, y = 1$,

$$x + xy = x$$
$$0 + 0 \cdot 1 = 0$$
$$0 + 0 = 0$$
$$0 = 0$$

3e cas: Pour $x = 1, y = 0$,

$$x + xy = x$$
$$1 + 1 \cdot 0 = 1$$
$$1 + 0 = 1$$
$$1 = 1$$

4e cas: Pour $x = 1, y = 1$,

$$x + xy = x$$
$$1 + 1 \cdot 1 = 1$$
$$1 + 1 = 1$$
$$1 = 1$$

Le théorème (14) peut également être démontré comme suit en mettant en facteur et en utilisant les théorèmes (6) et (2):

$$x + xy = x(1 + y)$$
$$= x \cdot 1 \quad \text{[en vertu du théorème (6)]}$$
$$= x \quad \text{[en vertu du théorème (2)]}$$

Tous ces théorèmes de Boole sont d'une grande utilité pour simplifier une expression logique, c'est-à-dire pour obtenir une expression comptant moins de termes. L'expression simplifiée permet de réaliser un circuit moins complexe que celui correspondant à l'expression originale. La plus grande partie du chapitre suivant est consacrée à la technique de la simplification ou minimisation des circuits. Pour le moment, les quelques exemples suivants vous montrent comment appliquer les théorèmes de l'algèbre booléenne.

EXEMPLE 3-13

Simplifiez l'expression $y = A\overline{B}D + A\overline{B}\,\overline{D}$.

Solution D'après le théorème (13) mettez en facteur $A\overline{B}$:

$$y = A\overline{B}(D + \overline{D})$$

En vertu du théorème (8), le terme entre parenthèses vaut 1, d'où

$$y = A\overline{B} \cdot 1$$
$$y = A\overline{B} \quad \text{[d'après le théorème (2)]}$$

EXEMPLE 3-14

Simplifiez $z = (\overline{A} + B)(A + B)$

Solution D'après le théorème (13), on peut développer cette expression en multipliant chacun des termes.

$$z = \overline{A} \cdot A + \overline{A} \cdot B + B \cdot A + B \cdot B$$

On sait, en vertu du théorème (4), que $\overline{A} \cdot A = 0$ et que $B \cdot B = B$ en vertu du théorème (3).

$$z = 0 + \overline{A} \cdot B + B \cdot A + B = \overline{A}B + AB + B$$

En mettant B en facteur [théorème (13)], il vient:

$$z = B(\overline{A} + A + 1)$$

Ce qui donne, en utilisant le théorème (6),

$$z = B$$

EXEMPLE 3-15

Simplifiez $x = ACD + \overline{A}BCD$

Solution En mettant en facteur CD, nous obtenons:

$$x = CD(A + \overline{A}B)$$

Nous savons, d'après le théorème (15) que $A + \overline{A}B$ égale $A + B$, d'où:

$$x = CD(A + B)$$
$$= ACD + BCD$$

QUESTIONS DE RÉVISION

1. Au moyen des théorèmes (13) et (14), simplifiez $y = A\overline{C} + AB\overline{C}$.
2. Au moyen des théorèmes (13) et (8), simplifiez $y = \overline{A}BC\overline{D} + \overline{A}B\overline{C}D$.

3-11 THÉORÈMES DE DE MORGAN

Deux des plus importants théorèmes de l'algèbre booléenne nous ont été légués par le mathématicien De Morgan. Les *théorèmes de De Morgan* se révèlent d'une grande utilité pour simplifier des expressions comprenant des sommes ou des produits de variables complémentés. Voici ces deux théorèmes:

(16) $\overline{(x + y)} = \overline{x} \cdot \overline{y}$

(17) $\overline{(x \cdot y)} = \overline{x} + \overline{y}$

Le théorème (16) affirme que la somme logique complémentée de deux variables est égale au produit logique des compléments de ces deux variables. De même, le théorème (17) stipule que le produit logique complémenté de deux variables est égal à la somme logique

des compléments de ces deux variables. La démonstration de ces deux théorèmes se fait simplement en considérant toutes les possibilités de x et de y. Nous laissons à l'étudiant le soin de faire cette démonstration dans un problème à la fin du chapitre.

Bien que ces théorèmes aient été formulés pour les variables simples x et y, ils demeurent aussi vrais pour les cas où x et y sont des expressions comprenant plusieurs variables. À titre d'illustration, appliquons ces théorèmes à l'expression $\overline{(A\overline{B} + C)}$:

$$\overline{(A\overline{B} + C)} = \overline{(A\overline{B})} \cdot \overline{C}$$

Notez que dans ce cas $x = A\overline{B}$ et $y = C$. Le résultat trouvé peut être simplifié une autre fois, car on y retrouve encore un produit logique complémenté, $A\overline{B}$. En vertu du théorème (17), il vient:

$$\overline{A\overline{B}} \cdot \overline{C} = (\overline{A} + \overline{\overline{B}}) \cdot \overline{C}$$

Comme $\overline{\overline{B}} = B$, le résultat définitif est alors:

$$(\overline{A} + B) \cdot \overline{C} = \overline{A}\overline{C} + B\overline{C}$$

Ce dernier résultat ne contient que des variables simples complémentées.

EXEMPLE 3-16

Simplifiez l'expression $z = \overline{(\overline{A} + C)} \cdot \overline{(B + \overline{D})}$ afin de n'avoir que des variables simples complémentées.

Solution D'après le théorème (17), on peut réécrire z comme suit:

$$z = \overline{(\overline{A} + C)} + \overline{(B + \overline{D})}$$

On peut analyser cette solution en supposant que l'on coupe en deux la grande barre d'inversion et que l'on remplace le signe ET ($\cdot$) par le signe OU ($+$). Maintenant, on peut simplifier $\overline{(\overline{A} + C)}$ et $\overline{(B + \overline{D})}$ en vertu du théorème (16). Ce qui donne:

$$\begin{aligned} z &= \overline{(\overline{A} + C)} + \overline{(B + \overline{D})} \\ &= (\overline{\overline{A}} \cdot \overline{C}) + \overline{B} \cdot \overline{\overline{D}} \end{aligned}$$

Dans cette dernière transformation, nous avons coupé en deux les deux grandes barres d'inversion et remplacé les signes ($+$) par des signes ($\cdot$). En éliminant les doubles complémentations, on obtient finalement:

$$z = A\overline{C} + \overline{B}D$$

L'exemple 3-16 nous montre que l'application des théorèmes de De Morgan permet de couper un signe de complémentation à n'importe quel point d'une expression et de changer l'opérateur en ce point par l'autre (changer $+$ par $\cdot$ ou vice versa). Cette démarche est répétée jusqu'à ce que l'expression ne soit plus formée que de variables simples complémentées. Voici deux exemples supplémentaires:

1.
$$\begin{aligned} z &= \overline{A + \overline{B} \cdot C} \\ &= \overline{A} \cdot \overline{(\overline{B} \cdot C)} \\ &= \overline{A} \cdot (\overline{\overline{B}} + \overline{C}) \\ &= \overline{A} \cdot (B + \overline{C}) \end{aligned}$$

2.
$$\begin{aligned} q &= \overline{(A + BC) \cdot (D + EF)} \\ &= \overline{(A + BC)} + \overline{(D + EF)} \\ &= (\overline{A} \cdot \overline{BC}) + (\overline{D} \cdot \overline{EF}) \\ &= [\overline{A} \cdot (\overline{B} + \overline{C})] + [\overline{D} \cdot (\overline{E} + \overline{F})] \\ &= \overline{A}\overline{B} + \overline{A}\overline{C} + \overline{D}\overline{E} + \overline{D}\overline{F} \end{aligned}$$

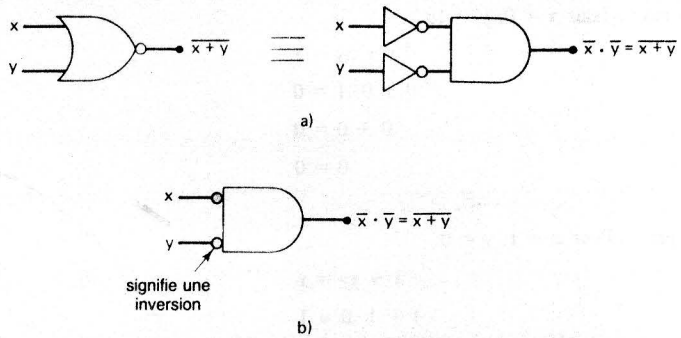


FIGURE 3-26 a) Circuits équivalents découlant directement du théorème (16); b) autre symbole de la fonction NI.

Les théorèmes de De Morgan sont étendus sans difficulté à plus de deux variables. Par exemple, il est possible de démontrer que:

$$\overline{x + y + z} = \overline{x} \cdot \overline{y} \cdot \overline{z}$$

$$\overline{x \cdot y \cdot z} = \overline{x} + \overline{y} + \overline{z}$$

et ainsi de suite pour un nombre de variables plus grand. De nouveau, nous tenons à vous rappeler que chaque variable peut être une expression formée de plusieurs variables.

Conséquences des théorèmes de **De Morgan** Examinons les théorèmes (16) et (17) du point de vue des circuits logiques. D'abord, voyons le théorème (16),

$$\overline{(x + y)} = \overline{x} \cdot \overline{y}$$

Le membre de gauche de cette équation peut être considéré comme la sortie d'une porte NI dont les entrées sont x et y. Par ailleurs, le membre de droite est l'expression décrivant d'abord l'inversion de x et y puis leur regroupement dans une porte ET. On a donc là deux représentions équivalentes, ce qu'illustre la figure 3-26 a). Ceci signifie qu'une porte ET ayant à chacune de ses entrées un INVERSEUR est équivalente à une porte NI. Dans les faits, la fonction NI peut être réalisée par l'une ou l'autre de ces représentions. Quand on utilise une porte ET avec des entrées inversées pour figurer la fonction NI, on a généralement recours au schéma montré à la figure 3-26 b), dans laquelle les petits ronds correspondent à des INVERSEURS.

Voyons maintenant le théorème (17):

$$\overline{x \cdot y} = \overline{x} + \overline{y}$$

Le membre de gauche de cette équation est réalisable au moyen d'une porte NON-ET dont les entrées sont x et y. Le membre de droite est réalisé en technique numérique, d'abord, en inversant les entrées x et y puis en les regroupant dans une porte OU. On peut voir à la figure 3-27 a) ces deux représentations équivalentes. La porte OU comprenant des INVERSEURS à chacune de ses entrées donne le même résultat que la porte NON-ET. Dans les faits, on utilise l'un ou l'autre de ces circuits pour obtenir la fonction NON-ET.

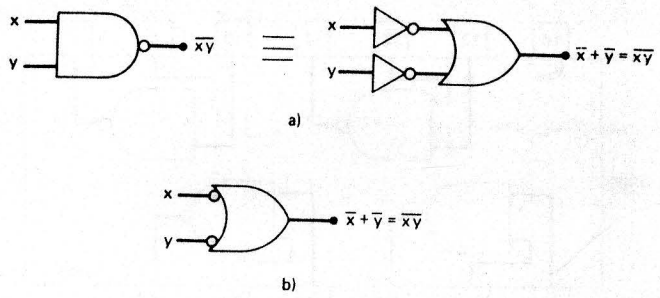


FIGURE 3-27 a) Circuits équivalents découlant directement du théorème (17); b) autre symbole de la fonction NON-ET.

Quand on utilise une porte OU ayant ses entrées inversées pour réaliser la fonction NON-ET, on a recours au schéma montré à la figure 3-27 b), dans laquelle les ronds représentent comme toujours l'opération d'inversion.

EXEMPLE 3-17

Réalisez avec une porte NON-ET et un INVERSEUR un circuit ayant comme expression de sortie $z = \overline{A} + \overline{B} + C$.

Solution Inversons deux fois le membre droit de l'expression. Cela est admissible, puisqu'une double complémentation ne modifie en rien la valeur logique:

$$z = \overline{\overline{\overline{A} + \overline{B} + C}}$$

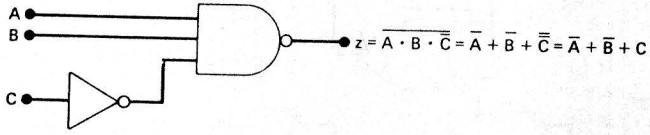


FIGURE 3-28 Exemple 3-17.

À présent au moyen du théorème de De Morgan, fractionnons la barre de surlignement inférieure:

$$z = \overline{\overline{A} \cdot \overline{\overline{B}} \cdot \overline{C}}$$

Annulons les doubles complémentations de A et de B, ce qui donne:

$$z = \overline{A \cdot B \cdot \overline{C}}$$

La sortie exprimée de cette façon nous montre immédiatement qu'une porte NON-ET peut être employée. Le circuit convenant est illustré à la figure 3-28.

QUESTIONS DE RÉVISION

1. Au moyen des théorèmes de De Morgan, transformez l'expression $z = \overline{(A + B) \cdot \overline{C}}$ en une expression formée exclusivement de variables simples complémentées.
2. Même chose que la question 1, mais cette fois-ci pour l'expression $y = \overline{R\overline{S}T + \overline{Q}}$.
3. Construisez, en utilisant seulement une porte NI et un INVERSEUR, un circuit ayant comme expression de sortie $z = \overline{A}\overline{B}C$.

3-12 UNIVERSALITÉ DES PORTES NON-ET ET DES PORTES NI

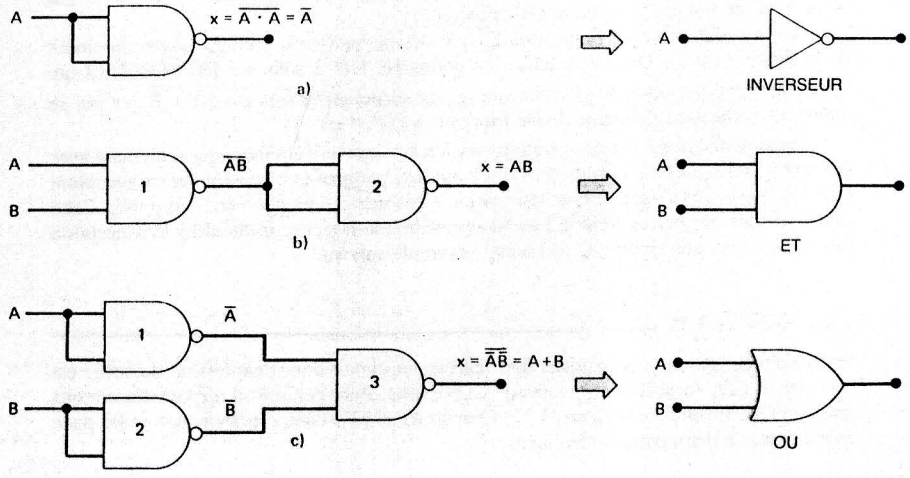
Toutes les expressions booléennes se résument à différentes combinaisons des opérations élémentaires OU, ET et INVERSION. Compte tenu de cela, il est donc possible de réaliser n'importe quelle expression en n'utilisant que des portes ET, des portes OU et des INVERSEURS. Par ailleurs, la matérialisation de *toute expression logique est faisable en ne recourant qu'à des portes* NON-ET, et à aucun autre type de porte. Il en est ainsi parce que les portes NON-ET, judicieusement câblées et regroupées, peuvent réaliser les opérations booléennes OU, ET et INVERSION. C'est ce que nous fait voir la figure 3-29.

Premièrement, la figure 3-29 a) nous montre une porte NON-ET dont les deux entrées sont connectées ensemble afin que le même signal A leur soit appliqué. Utilisée de cette manière, la porte NON-ET sert simplement d'INVERSEUR, puisque son résultat de sortie est $x = \overline{A \cdot A} = \overline{A}$.

À la figure 3-29 b), deux portes NON-ET permettent de réaliser l'opération ET. La seconde porte NON-ET agit comme un INVERSEUR pour changer $\overline{AB}$ en $\overline{\overline{AB}} = AB$, ce qui donne la fonction ET recherchée.

L'opération OU peut être matérialisée au moyen de portes NON-ET en procédant comme à la figure 3-29 c). Dans cette figure, les portes NON-ET 1 et 2 réalisent l'IN-

FIGURE 3-29 Réalisation des trois opérations booléennes au moyen de portes NON-ET.



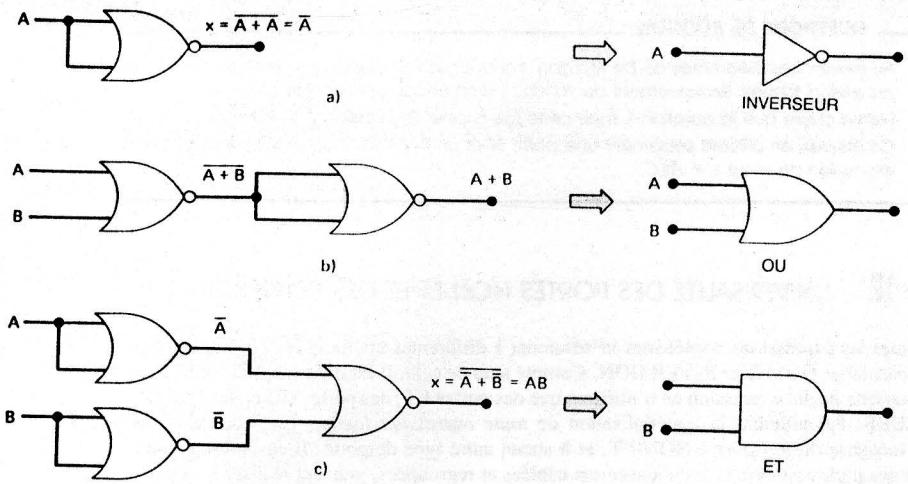


FIGURE 3-30 Réalisation des trois opérations booléennes au moyen de portes NI.

VERSION des entrées, ce qui donne en sortie $x = \overline{\bar{A} \cdot \bar{B}}$ que l'on sait être égal, d'après un théorème de De Morgan, à $x = A + B$.

Une démarche analogue nous permet de montrer que des portes NI associées de la bonne manière permettent de réaliser n'importe laquelle des opérations booléennes. C'est d'ailleurs ce que nous fait voir la figure 3-30. La partie a) montre comment une porte NI devient un INVERSEUR si on raccorde ensemble ses entrées, puisqu'alors le signal de sortie est $x = \overline{A + A} = \bar{A}$.

À la figure 3-30 b), deux portes NI sont associées de manière à produire l'opération OU. La seconde porte NI agit comme INVERSEUR pour changer $\overline{A + B}$ en $\overline{\overline{A + B}} = A + B$, ce qui est la fonction OU recherchée.

La matérialisation de l'opération ET par des portes NI est faite au moyen du circuit de la figure 3-30 c). Dans ce circuit, les portes NI 1 et 2 sont des INVERSEURS qui complémentent les entrées, de façon que le signal de sortie soit $x = \overline{\bar{A} + \bar{B}}$, ce qui se réduit, en vertu d'un théorème de De Morgan, à $x = A \cdot B$.

Étant donné que les trois opérations booléennes peuvent être réalisées seulement avec des portes NON-ET, il s'ensuit que tout circuit logique peut être construit en ne recourant qu'à des portes NON-ET. Cette affirmation tient également pour les portes NI. Cette caractéristique des portes NON-ET et NI se révèle d'une grande utilité dans la conception des circuits logiques comme le démontre l'exemple suivant.

EXEMPLE 3-18

Un concepteur de circuits logiques doit réaliser un circuit dont l'expression de sortie est $x = AB + CD$, en utilisant le moins de CI possible. Pour ce faire, il dispose des circuits intégrés TTL illustrés à la figure 3-31. Chacun de ces CI comporte dans une seule puce quatre portes à deux entrées identiques.

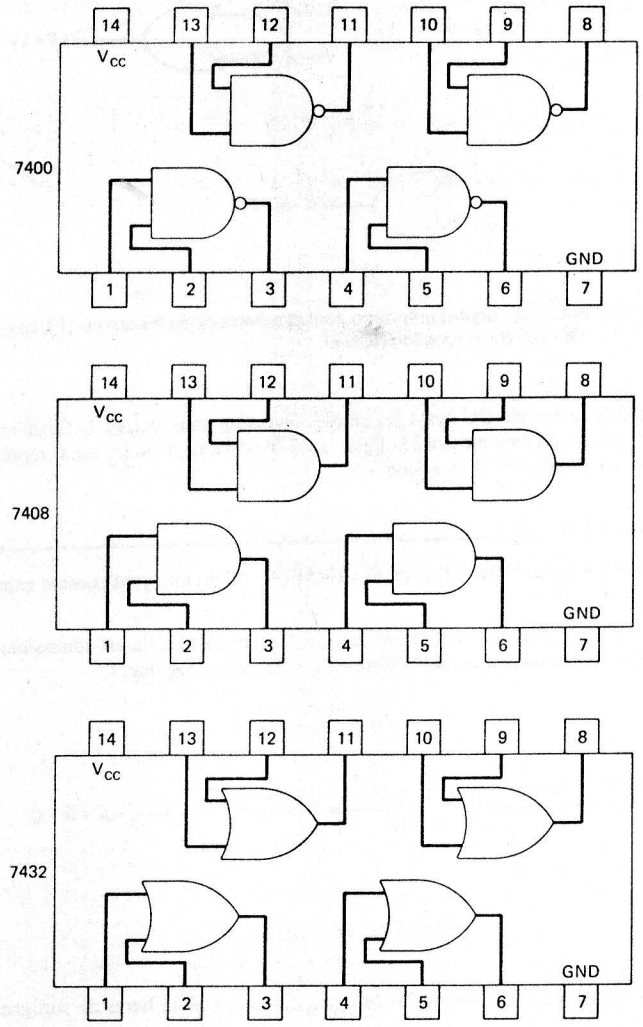


FIGURE 3-31 Les CI dont dispose le concepteur de l'exemple 3-18.

La solution directe à ce problème est d'utiliser deux portes ET et une porte OU [schéma de la figure 3-32 a)]. Pour construire ce circuit, il faut utiliser les CI 7408 et 7432; en outre, sur ces deux puces des portes sont gaspillées puiqu'elles ne servent pas.

Une autre façon possible de réaliser l'expression donnée consiste à prendre le circuit de la figure 3-32 a) et à remplacer chaque porte ET et la porte OU par la combinaison équivalente faite de portes NON-ET, illustrée à la figure 3-29. Ce qui donne le circuit de la figure 3-32 b).

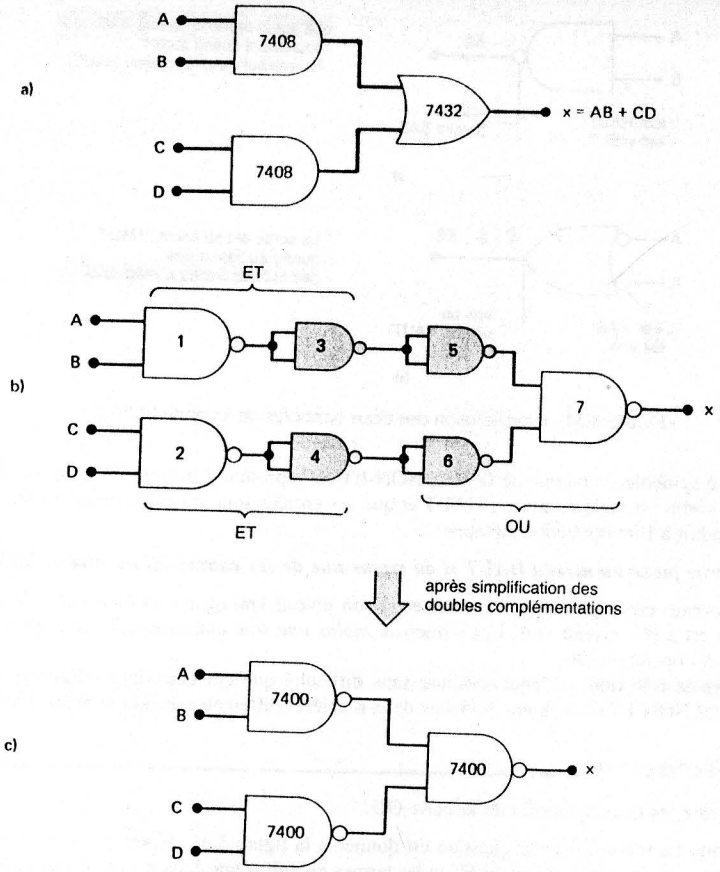


FIGURE 3-32 Diverses solutions possibles du problème de l'exemple 3-18.

À prime abord, on observe que ce nouveau circuit nécessite sept portes NON-ET. Toutefois, comme les portes NON-ET 3 et 5 sont des INVERSEURS raccordés en série, on peut les éliminer du circuit puisqu'ils effectuent une double complémentation du signal de sortie de la porte NON-ET 1. Pour la même raison, on enlève les deux portes NON-ET 4 et 6. Le circuit définitif, après élimination des couples d'INVERSEURS, est celui de la figure 3-32 c).

Ce dernier circuit est plus économique que celui de la figure 3-32 a), car il emploie trois portes NON-ET provenant du même CI, le 7400.

QUESTIONS DE RÉVISION

1. De combien de manières pouvons-nous maintenant réaliser l'opération d'inversion dans un circuit logique?

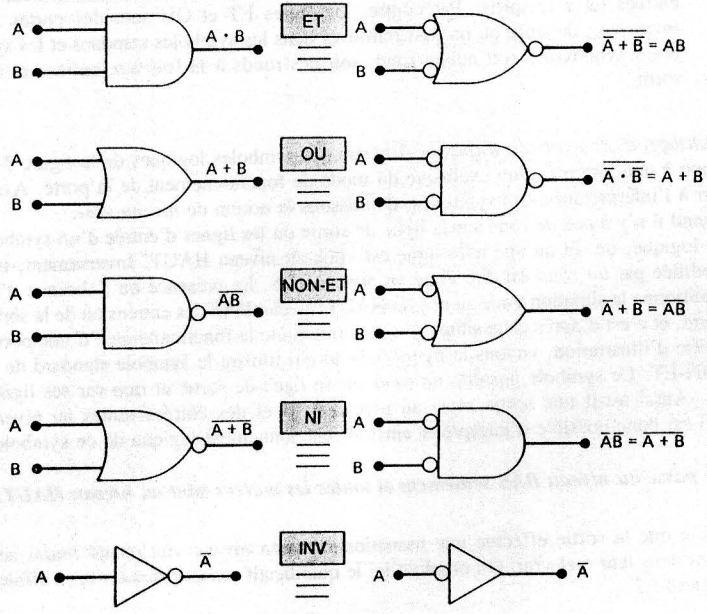
2. Matérialisez l'expression $x = (A + B)(C + D)$ en recourant à des portes OU et ET. Ensuite, obtenez cette expression au moyen de portes NI en remplaçant chaque porte OU et ET par leur combinaison équivalente faite de portes NI, illustrée à la figure 3-30. Quel circuit est le plus économique?
3. Écrivez l'expression de sortie du circuit de la figure 3-32 c) puis utilisez les théorèmes de De Morgan pour prouver que cette expression est équivalente à celle du circuit de la figure 3-32 a).

3-13 REPRÉSENTATIONS SYNONYMES DES PORTES LOGIQUES

Nous vous avons exposé les cinq portes logiques élémentaires (ET, OU, INVERSEUR, NON-ET et NI) et illustré les symboles standard qu'on emploie pour les représenter dans les schémas de circuits logiques. Il se peut que vous rencontriez de nombreux schémas constitués exclusivement des symboles standard, comme il se peut (et c'est de plus en plus courant) que vous ayez à travailler avec des diagrammes sur lesquels on retrouve *outre* les symboles standard un ensemble de symboles synonymes.

Avant d'invoquer les raisons qui justifient l'emploi d'un symbole synonyme pour une porte logique, nous allons vous présenter le symbole synonyme de chaque porte et montrer que ces symboles sont équivalents aux symboles standard. Reportez-vous à la figure 3-33. Le côté gauche de cette figure contient les symboles standard des portes logiques, tandis

FIGURE 3-33 Symboles standard et symboles synonymes des diverses portes logiques et de l'inverseur.



que le côté droit est constitué des symboles synonymes. Le symbole synonyme de chaque porte découle du symbole standard auquel on fait subir les transformations suivantes:

1. Inverser toutes les entrées et la sortie du symbole standard. Pour ce faire, ajouter des ronds aux lignes d'entrée et de sortie qui en sont dépourvues et enlever les ronds des lignes qui en ont.
2. Sauf pour l'INVERSEUR, changer l'opérateur ET pour OU, ou bien OU pour ET.

Par exemple, le symbole standard d'une porte NON-ET est un symbole ET ayant un rond dans sa ligne de sortie. Pour obtenir son symbole synonyme, suivre les étapes 1 et 2 ci-dessus: enlevez le rond de la sortie et mettez un rond à chacune des entrées, puis changez ET pour OU. Le résultat est un symbole OU avec des ronds aux entrées.

Les théorèmes de De Morgan permettent de démontrer sans difficulté que ce symbole synonyme est équivalent au symbole standard, car on sait que les ronds représentent l'opération d'inversion. L'expression de sortie du symbole NON-ET standard est $\overline{AB} = \overline{A} + \overline{B}$, et c'est la même que celle du symbole synonyme. On peut répéter la même analyse pour chaque pair de symboles de la figure 3-33.

Nous tenons à souligner quelques points concernant l'équivalence des symboles logiques:

1. L'équivalence demeure vraie *quel que soit le nombre* d'entrées de la porte.
2. Aucun symbole standard n'a de rond sur ses entrées, tandis que tous les symboles synonymes en ont sur toutes leurs entrées.
3. Le symbole standard et le symbole synonyme d'une même porte correspondent au même circuit physique; il n'y a aucune différence matérielle entre les circuits représentés par ces deux symboles.
4. Les portes NI et NON-ET sont des portes inversives, de sorte qu'on doit retrouver dans les symboles standard et les symboles synonymes des ronds seulement aux entrées *ou* à la sortie. Par contre, les portes ET et OU sont des portes *non inversives*, de sorte qu'on doit retrouver dans les symboles standard et les symboles synonymes soit aucun rond, soit des ronds à la fois aux entrées et à la sortie.

Interprétation d'un symbole logique Chacun des symboles logiques de la figure 3-33 correspond à une interprétation exclusive du mode de fonctionnement de la porte. Avant de passer à l'interprétation des symboles, définissons la notion de *niveau vrai*.

Quand il n'y a *pas de rond* sur la ligne de sortie ou les lignes d'entrée d'un symbole de porte logique, on dit qu'une telle ligne est vraie au niveau HAUT. Inversement, une ligne modifiée par *un rond* est dite vraie au niveau BAS. La présence ou l'absence d'un rond conditionne la situation vraie au niveau HAUT/niveau BAS des entrées ou de la sortie d'une porte, et c'est d'après cette situation qu'on interprète le fonctionnement d'une porte.

À titre d'illustration, voyons la figure 3-34 a) qui illustre le symbole standard de la porte NON-ET. Ce symbole possède un rond sur sa ligne de sortie et rien sur ses lignes d'entrées. Aussi a-t-il une sortie vraie au niveau BAS et des entrées vraies au niveau HAUT. Il est donc possible d'interpréter ainsi le fonctionnement logique de ce symbole:

La sortie passe au niveau BAS seulement si toutes les entrées sont au niveau HAUT.

Ceci signifie que la sortie effectue une transition vers son niveau vrai quand *toutes* ses entrées sont dans leur état vrai. On emploie ici le qualificatif *toutes* parce qu'on a affaire à un opérateur ET.

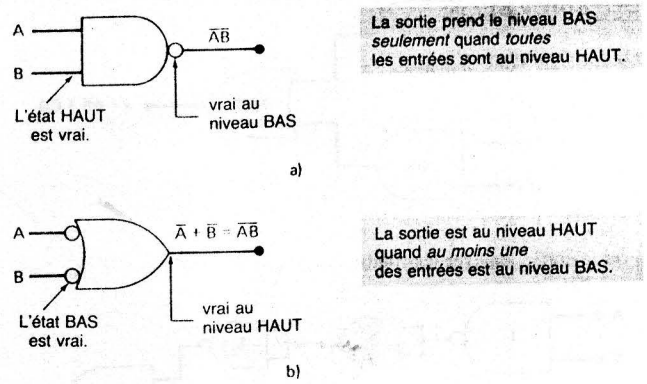


FIGURE 3-34 Interprétation des deux symboles de la porte NON-ET.

Le symbole synonyme de la porte NON-ET est reproduit à la figure 3-34 b); on voit que sa sortie est vraie au niveau HAUT et que ses entrées sont vraies au niveau BAS, ce qui conduit à l'interprétation suivante:

La sortie passe au niveau HAUT si au moins une de ses entrées est au niveau BAS.

De nouveau, ceci signifie que la sortie est à son niveau vrai quand *au moins une* de ses entrées est à son niveau vrai. Les termes *au moins une* sont employés ici parce qu'on a affaire à l'opération OU.

Après réflexion, on peut conclure sans difficulté que ces deux interprétations des symboles NON-ET de la figure 3-34 sont deux manières différentes de dire la même chose.

EXEMPLE 3-19

Interprétez les deux symboles de la porte OU.

Solution La réponse à cette question est donnée à la figure 3-35. Notez qu'on utilise le mot *toutes* au sujet de l'opérateur ET et les termes *au moins une* dans le cas de l'opérateur OU.

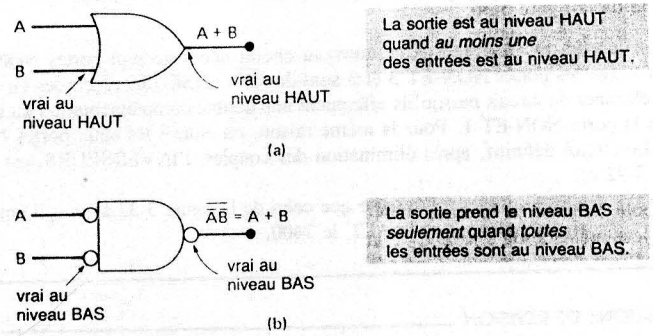


FIGURE 3-35 Interprétation des deux symboles de la porte OU.

RÉCAPITULATION

À ce point-ci, vous vous demandez sans doute pourquoi on a deux symboles différents pour chaque porte logique et deux interprétations différentes. Ne vous en faites pas, nous vous expliquerons les raisons qui justifient cet état de fait dans la section suivante. Pour le moment, résumons les principaux résultats concernant les représentations des portes logiques.

1. Pour trouver le symbole synonyme d'une porte logique, partir du symbole standard et changer l'opérateur (ET pour OU ou OU pour ET) puis, pour les entrées et la sortie, enlever les ronds s'il y en a ou en mettre s'il n'y en a pas.
2. Pour interpréter le fonctionnement d'une porte logique, relever d'abord l'état vrai des entrées et l'état vrai de la sortie, soit 0 ou 1. Ensuite, prendre conscience que le niveau vrai de la sortie se produit quand toutes les entrées sont à leur niveau vrai, dans le cas de l'opérateur ET, ou quand au moins une entrée est à son niveau vrai, dans le cas de l'opérateur OU.

QUESTIONS DE RÉVISION

1. Interprétez le fonctionnement du symbole standard de la porte NI de la figure 3-33.
2. Même question pour le symbole synonyme de la porte NI.
3. Même question pour le symbole synonyme de la porte ET.
4. Même question pour le symbole standard de la porte ET.

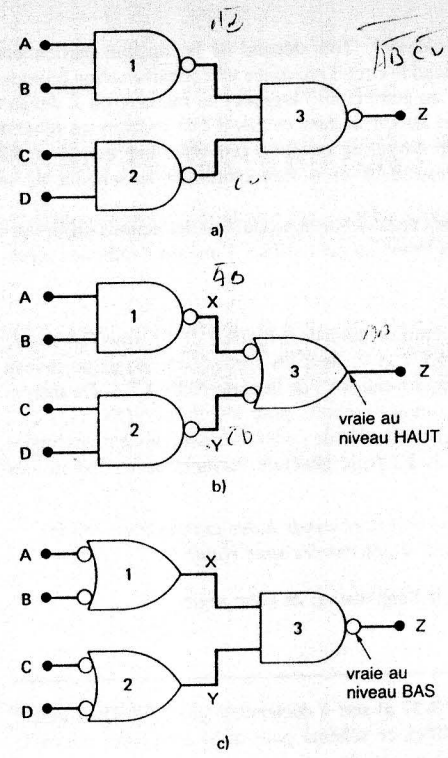
3-14 QUELLE REPRÉSENTATION DE PORTE DOIT-ON CHOISIR?

Certains concepteurs et de nombreux livres de référence n'utilisent que les symboles standard des portes logiques dans leurs schémas. Bien qu'il n'y ait rien de répréhensible à cela, ce n'est pas la façon la plus simple de suivre le fonctionnement du circuit. Une utilisation judicieuse des symboles des portes synonymes dans un schéma de circuit peut rendre beaucoup plus clair le fonctionnement du circuit. C'est d'ailleurs ce que nous montre l'exemple de la figure 3-36.

Le circuit de la figure 3-36 a) est formé de trois portes NON-ET raccordées afin de produire une sortie Z qui est fonction de A, B, C, D. Dans ce schéma, les portes NON-ET sont représentées au moyen de leur symbole standard. Bien que logiquement exact, ce schéma n'est pas celui qui nous aide le mieux à comprendre les fonctions du circuit. Les représentations améliorées des figures 3-36 b) et c) rendent possible une analyse plus facile du fonctionnement du circuit.

Ainsi, la représentation de la figure 3-36 b) est déduite de l'originale en substituant au symbole standard de la porte NON-ET 3 son symbole synonyme. Dans ce schéma, la sortie Z est prélevée du symbole de la porte NON-ET dont la sortie est vraie au niveau HAUT. C'est donc dire que Z prend le niveau HAUT seulement quand X ou Y est au niveau BAS. Comme X et Y sont les sorties de symboles NON-ET qui sont vraies au niveau BAS, il vient que X sera à 0 seulement si $A = B = 1$, et Y sera à 0 seulement si $C = D = 1$. Sommairement, le fonctionnement de ce circuit peut donc être décrit ainsi.

La sortie Z est au niveau HAUT quand A = B = 1 ou quand C = D = 1.



A	B	C	D	Z
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

d)

FIGURE 3-36 a) Circuit original réalisé avec des symboles standard de porte NON-ET; b) représentation équivalente dans le cas où la sortie z est vraie au niveau HAUT; c) représentation équivalente dans le cas où la sortie z est vraie au niveau BAS; d) table de vérité.

Cette affirmation peut se traduire sous la forme d'une table de vérité dans laquelle $Z = 1$ toutes les fois que $A = B = 1$ ou que $C = D = 1$, et $Z = 0$ toutes les autres fois. Cette table est montrée à la figure 3-36 d).

La représentation de la figure 3-36 c) se déduit aussi du schéma original en substituant aux symboles des portes NON-ET 1 et 2 leurs symboles synonymes. Dans cette représentation équivalente, la sortie Z est prélevée de la porte NON-ET dont la sortie est vraie au niveau BAS. On peut donc conclure que Z prend la valeur 0 seulement quand $X = Y = 1$. Puisque les sorties X et Y sont toutes les deux vraies au niveau HAUT, il vient que X est égal à 1 quand A ou B est 0, et que Y est égal à 1 quand C ou D est 0. Sommairement, le fonctionnement de ce circuit peut se traduire ainsi.

La sortie Z est au niveau BAS quand A ou B est au niveau BAS et quand C ou D est au niveau BAS.

Cette affirmation peut se traduire sous la forme d'une table de vérité dans laquelle $Z = 0$ toutes les fois que A ou B est 0 en même temps que C ou D est 0. Toutes les autres fois $Z = 1$. La table de vérité résultante est la même que celle obtenue pour le schéma de circuit de la figure 3-36 b).

Quel schéma de circuit doit-on employer? Tout dépend de la fonction particulière réalisée par la sortie du circuit. Ainsi, quand le circuit provoque une certaine action (comme allumer une lumière, mettre en marche un autre circuit logique) au moment où Z devient égal à 1, on dit que Z est vrai au niveau HAUT et dans ce cas il faut recourir au schéma de la figure 3-36 b). En revanche, si le circuit en question provoque une action quand Z devient 0, alors Z est vrai au niveau BAS et il faut employer le schéma de la figure 3-36 c).

Évidemment, dans les cas où les deux états de sortie produisent des actions différentes indispensables, et qu'il y a deux niveaux vrais, on peut utiliser l'une ou l'autre des représentations du circuit.

Positionnement du rond Examinez la représentation de la figure 3-36 b), vous remarquez que les symboles pour les portes NON-ET 1 et 2, dont les sorties sont vraies au niveau BAS, concordent avec les entrées vraies au niveau BAS de la porte NON-ET 3. De même, la représentation de la figure 3-36 c) a comme symboles pour les portes NON-ET 1 et 2 ceux qui ont leur sortie vraie au niveau HAUT afin de concorder avec les entrées vraies au niveau HAUT de la porte NON-ET 3. La règle générale suivante découle donc des constatations précédentes:

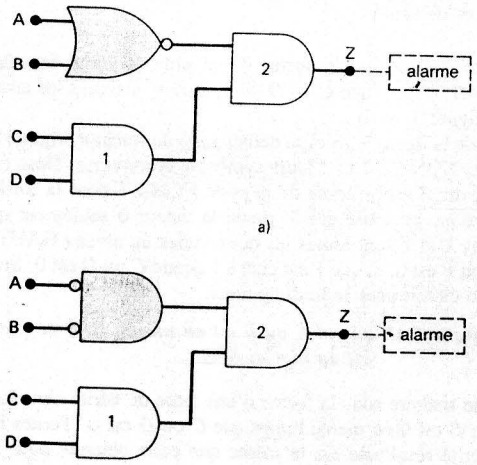
Autant que possible, connecter des sorties avec ronds à des entrées avec ronds, et des sorties sans rond à des entrées sans rond.

Les exemples suivants vous éclaireront sur l'application de cette règle.

EXEMPLE 3-20

Le circuit logique schématisé à la figure 3-37 a) sert à déclencher une ALARME quand la sortie Z passe au niveau HAUT. Modifiez ce schéma pour qu'il représente mieux le fonctionnement du circuit.

FIGURE 3-37 Exemple 3-20.



Solution Puisqu'une alarme est en marche quand $Z = 1$, Z est vrai au niveau HAUT. C'est donc dire que le symbole de la porte ET 2 n'a pas à être modifié. Par contre, le symbole de la porte NI doit être remplacé par son symbole synonyme pour qu'une sortie sans rond (vrai au niveau HAUT) concorde avec l'entrée sans rond de la porte ET 2. La réponse est le schéma de la figure 3-37 b).

EXEMPLE 3-21

Quand la sortie du circuit logique de la figure 3-38 a) passe au niveau BAS, cela met en marche un autre circuit logique. Modifiez ce schéma de circuit pour qu'il représente mieux le fonctionnement du circuit.

Solution Étant donné que Z est vrai au niveau BAS, il faut substituer au symbole de la porte OU 2 son symbole synonyme, comme le montre la figure 3-38 b). Ce nouveau symbole possède des ronds sur ses entrées, d'où la nécessité de changer les symboles de la porte ET et de la porte OU 1 pour avoir des sorties avec ronds (voir figure 3-38 b). L'INVERSEUR a déjà une sortie avec un rond.

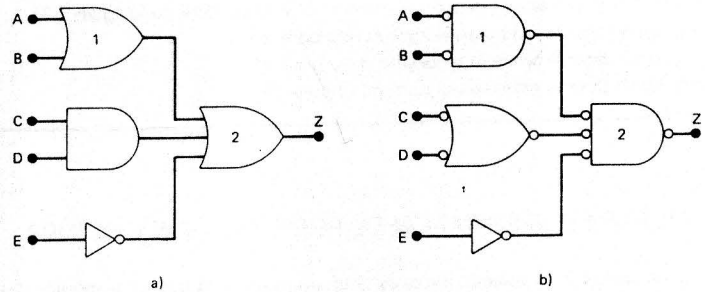


FIGURE 3-38 Exemple 3-21.

Analyse des circuits Quand un schéma de circuit logique est tracé en respectant les règles suivies dans ces exemples, il est beaucoup plus facile pour l'ingénieur, le technicien ou l'étudiant de suivre le cheminement du signal dans le circuit et de déterminer les conditions d'entrée qui font apparaître la bonne valeur de sortie. C'est ce que se proposent d'illustrer les exemples suivants qui, incidemment, utilisent des schémas de circuits d'un vrai micro-ordinateur.

EXEMPLE 3-22

Le circuit logique de la figure 3-39 génère une sortie, *MEM*, qui sert à rendre actives les puces mémoires d'un certain micro-ordinateur. Trouvez les conditions d'entrée *MEM*.

Solution Une manière de procéder consiste à exprimer *MEM* en fonction de *LE*, *MORTE-A*, *MORTE-B* et *VIVE* et de passer en revue les 16 combinaisons possibles des entrées. Cette marche à suivre donne la réponse, mais exige beaucoup plus de temps qu'il n'est nécessaire.

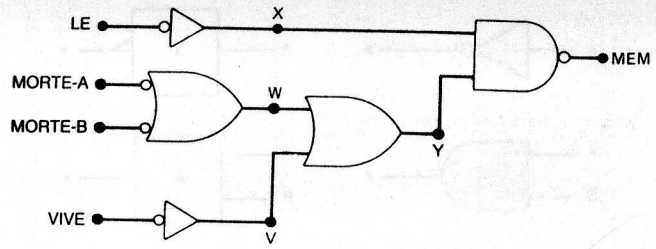


FIGURE 3-39 Exemple 3-22.

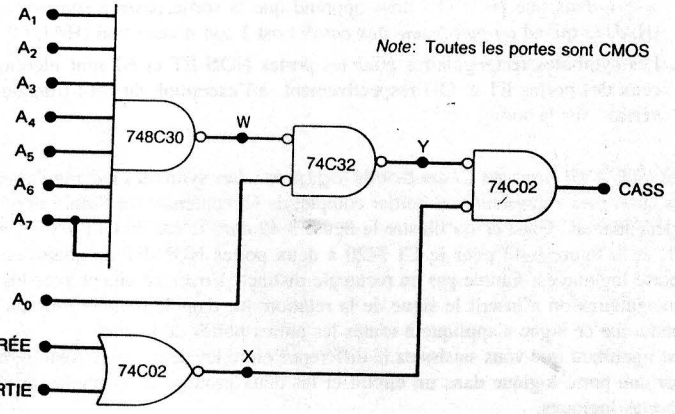
Une méthode beaucoup plus rapide est l'interprétation du schéma en recourant aux idées présentées dans les deux dernières sections. Voici les étapes de cette méthode:

1. *MEM* est vrai au niveau BAS et devient 0 seulement quand *X* et *Y* sont à 1.
2. *X* est à 1 seulement quand $LE = 0$.
3. *Y* est à 1 seulement quand *W* ou *V* est à 1.
4. *V* est à 1 quand $VIVE = 0$.
5. *W* est à 1 quand *MORTE-A* ou *MORTE-B* $= 0$.
6. En réunissant toutes ces conditions, on voit que *MEM* est au niveau BAS quand $LE = 0$ *et* au moins une des trois entrées *MORTE-A*, *MORTE-B* ou *VIVE* est au niveau BAS.

EXEMPLE 3-23

Le circuit logique de la figure 3.40 commande le moteur d'un enregistreur à cassette quand le micro-ordinateur envoie des données à cet enregistreur ou en reçoit de lui. Ce circuit met en marche le moteur quand $CASS = 1$. Établissez les conditions d'entrée qui mettent le moteur en marche.

FIGURE 3-40 Exemple 3-23



Solution À nouveau, interprétons le schéma en procédant étape par étape.

1. *CASS* est vrai au niveau HAUT et prend la valeur 1 seulement quand $X = Y = 0$.
2. *X* est égal à 0 quand *ENTRÉE* ou *SORTIE* est à 1.
3. *Y* est égal à 0 quand $W = 0$ et $A_0 = 0$.
4. *W* est égal à 0 seulement quand les lignes A_1 à A_7 sont toutes égales à 1.
5. En réunissant toutes ces conditions, on voit que *CASS* est à 1 quand $A_1 = A_2 = A_3 = A_4 = A_5 = A_6 = A_7 = 1$ et $A_0 = 0$ et quand *ENTRÉE* ou *SORTIE* ou les deux sont à 1.

Notez le symbole surprenant pour la porte NON-ET à huit entrées (74C30) et le raccordement du signal A_7 à deux broches de cette porte.

RÉCAPITULATION

Les exemples présentés dans cette section vous ont montré comment tracer un schéma de circuit pour que le fonctionnement du circuit soit le plus explicite possible. Le choix du symbole qui convient pour chaque porte logique simplifie l'analyse du schéma du circuit visant à établir les conditions d'entrée qui produisent la sortie recherchée.

Tout au long de ce livre nous utiliserons cette méthode de présentation pour les schémas des circuits. Toutefois, pour toutes sortes de raisons, vous ne retrouverez dans certains schémas que des symboles standard.

Niveaux valides Nous avons décrit les signaux logiques comme étant vrais au niveau BAS ou vrais au niveau HAUT. Par exemple, la sortie *MEM* de la figure 3-39 est vraie au niveau BAS, et la sortie *CASS* de la figure 3-40 est vraie au niveau HAUT, puisque ce sont ces niveaux qui déclenchent une action. Pour la même raison, les entrées A_1 à A_7 sont vraies au niveau HAUT et l'entrée A_0 est vraie au niveau BAS.

Quand un signal logique est dans son état vrai, on dit qu'il est **valide**. Par exemple, quand on dit que A_0 est valide, on veut signifier qu'il est à son niveau vrai. Quand un signal n'est pas à son niveau vrai, on dit qu'il est **non valide**. Ainsi, quand on affirme que *CASS* est non valide, on veut rapporter qu'il n'est pas dans son niveau vrai (il est à ZÉRO).

Il découle donc de ceci que les expressions « valide » et « non-valide » sont respectivement des synonymes de « niveau vrai » et de « niveau non vrai »:

valide = niveau vrai
non-valide = niveau non vrai

Ces deux groupes de termes sont utilisés fréquemment en technologie numérique; vous devez pour cette raison connaître les deux façons de décrire l'état actif d'un signal logique.

Désignation des signaux logiques vrais au niveau BAS Il est devenu pratique courante de surmonter les signaux d'une barre pour indiquer qu'ils sont vrais au niveau bas. La barre de surlignement est donc une marque signalant que le signal est vrai au niveau BAS; évidemment, l'absence de barre indique que le signal est vrai au niveau HAUT.

À titre d'exemple, voyez la figure 3-39 où tous les signaux sont vrais au niveau BAS; ils pourraient tous être désignés ainsi:

$\overline{RD}$, $\overline{MORTE\text{-}A}$, $\overline{MORTE\text{-}B}$, $\overline{VIVE}$, $\overline{MEM}$

Rappelez-vous que la barre de surlignement est une façon commode de signaler que les signaux sont vrais au niveau BAS. Nous respecterons cette convention de désignation des signaux logiques partout où ce sera utile.

QUESTIONS DE RÉVISION

1. Au moyen de la méthode exposée dans les exemples 3-22 et 3-23, trouvez les conditions d'entrée qui placent sur la sortie du circuit de la figure 3-37 b) la valeur de déclenchement.
2. Même question pour le circuit de la figure 3-38 b).
3. Combien y a-t-il de portes NON-ET sur la figure 3-39?
4. Combien y a-t-il de portes NI sur la figure 3-40?
5. Quel est le niveau logique de sortie du circuit de la figure 3-38 b) quand toutes les entrées sont valides?
6. Quelles doivent être les entrées pour que soit validée la sortie du système d'alarme de la figure 3-37 b)?

3-15 SYMBOLES LOGIQUES NORMALISÉS IEEE/ANSI

Les symboles logiques que nous avons utilisés depuis le début de ce livre sont les symboles courants bien connus en usage depuis des années dans l'industrie des techniques numériques. Ces symboles conviennent très bien aux portes logiques de base du fait que chaque symbole a une forme différente, et que chaque entrée de porte joue le même rôle. Cependant, ces symboles ne véhiculent pas suffisamment d'informations quand il s'agit de représenter des dispositifs logiques plus complexes comme les bascules, les compteurs, les décodeurs, les multiplexeurs, les mémoires et les CI raccordés aux microprocesseurs. Ces CI complexes ont souvent plusieurs entrées et plusieurs sorties, chacune ayant des fonctions et des modes opératoires différents.

Afin de mieux représenter les circuits logiques complexes, un nouvel ensemble de symboles normalisés a été divulgué en 1984 dans la norme IEEE/ANSI 91-1984. Ces nouveaux symboles sont progressivement adoptés par de plus en plus d'entreprises en électronique et de fabriquants de circuits intégrés, certains mêmes ont commencé à les utiliser dans leur documentation imprimée. En outre, aux États-Unis, les forces armées exigent dans tous leurs contrats que l'on utilise ces symboles. Il est donc important de se familiariser avec cette nouvelle norme, puisqu'à terme, après un processus qui sera toutefois très long, les symboles proposés remplaceront les symboles habituels.

La principale particularité de cette nouvelle norme est qu'elle propose pour tous les dispositifs un symbole rectangulaire, plutôt que des symboles de formes différentes. Le rapport entre les entrées et les sorties est signalé au moyen d'une notation spéciale. La figure 3-41 montre les symboles traditionnels et en vis-à-vis les nouveaux symboles rectangulaires pour les portes de base. Étudiez-les attentivement et notez les points suivants:

1. Dans les nouveaux symboles on retrouve un petit triangle droit ◺ à la place du petit rond des symboles traditionnels. Comme ce rond, le triangle signifie une inversion du niveau logique. La présence ou l'absence de triangle, tout comme d'un rond, qualifie une entrée ou une sortie vraie au niveau BAS ou au niveau HAUT.
2. Les signes de la notation spéciale inscrits dans chaque symbole rectangulaire correspondent à une relation entre les entrées et la sortie. Le signe « 1 » de

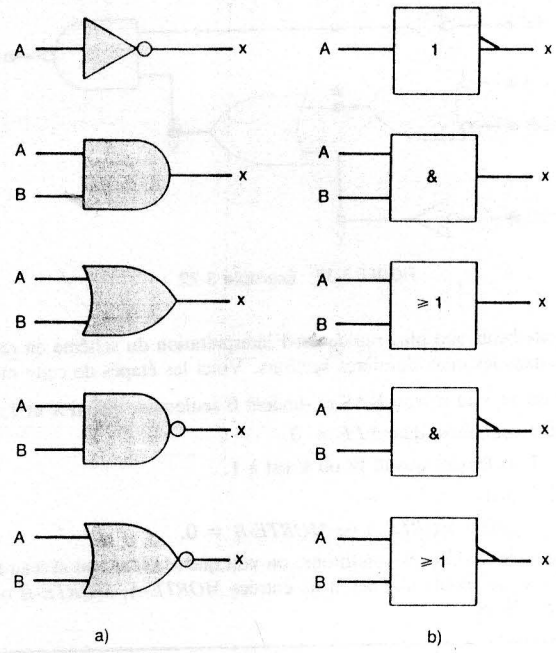


FIGURE 3-41 Symboles logiques standard a) traditionnels; b) rectangulaires.

l'INVERSEUR désigne un dispositif à *une* entrée; le triangle sur la sortie informe que la sortie sera vraie au niveau BAS quand cette entrée seule sera vraie au niveau HAUT.

Le signe « & » du symbole d'une porte ET signifie que la sortie est vraie au niveau HAUT quand toutes les entrées sont vraies au niveau HAUT. Le signe « ≥ » dans une porte OU nous apprend que la sortie passe à son niveau vrai (HAUT) quand *au moins une des entrées* est à son niveau vrai (HAUT).

3. Les symboles rectangulaires pour les portes NON-ET et NI sont identiques à ceux des portes ET et OU respectivement, à l'exception du petit triangle d'inversion sur la sortie.

Symboles IEEE/ANSI pour les CI de portes logiques Les symboles rectangulaires sont utilisables aussi pour représenter un boîtier complet de CI contenant un certain nombre de portes indépendantes. C'est ce qu'illustre la figure 3-42 dans le cas du CI à six inverseurs 7404 TTL et la figure 3-43 pour le CI 7420 à deux portes NON-ET de quatre entrées. Chaque porte logique est figurée par un rectangle distinct. Voyez comment avec les symboles rectangulaires on n'inscrit le signe de la relation que dans le premier bloc du haut, sous-entendu que ce signe s'applique à toutes les autres portes de la puce.

Il est important que vous saisissiez la différence entre les deux façons synonymes de représenter une porte logique dans un circuit et les deux groupes de symboles différents pour les portes logiques.

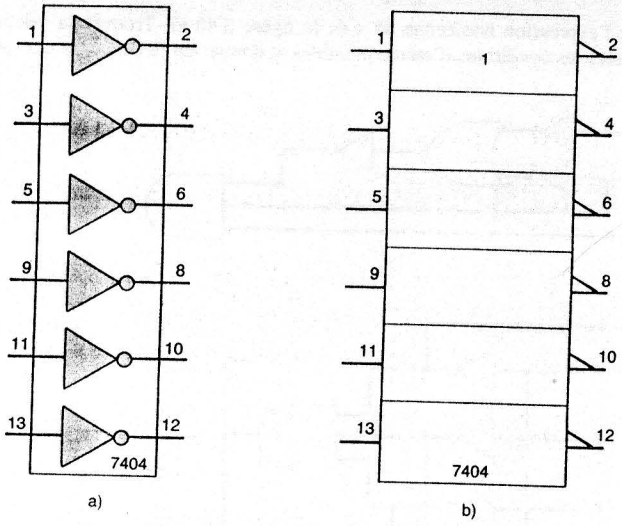


FIGURE 3-42 CI 7404 à six INVERSEURS: a) symbole traditionnel; b) symbole rectangulaire. Le libellé « 1 » dans le premier rectangle vaut aussi pour tous les autres blocs.

Vous choisissez l'ensemble de symboles logiques (logigrammes) normalisés qui vous convient — l'ensemble traditionnel (chaque symbole a une forme distinctive) ou le nouvel ensemble rectangulaire. Quels que soient les symboles que vous retenez, il y a deux façons équivalentes de décrire une porte dans un schéma logique suivant son niveau de sortie vraie. L'exemple 3-24 illustre ceci

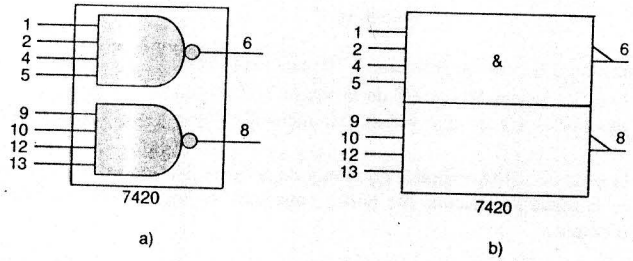


FIGURE 3-43 Le CI 7420 à deux portes NON-ET de quatre entrées: a) le symbole traditionnel; b) le symbole rectangulaire.

EXEMPLE 3-24

La figure 3-44 a) nous fait voir deux représentations d'une porte NI réalisées en se servant des symboles logiques traditionnels. Rappelez-vous que la représentation utilisée dans un

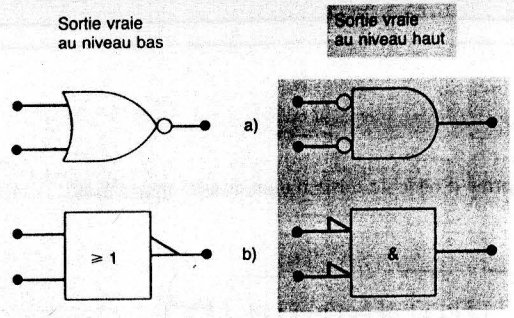


FIGURE 3-44 Les deux représentations synonymes d'une porte NI obtenues a) avec les symboles traditionnels; b) avec les symboles rectangulaires.

schéma de circuit est conditionnée par la valeur du niveau vrai de la sortie. Redessinez ces deux représentations mais en utilisant cette fois-ci les nouveaux symboles IEEE/ANSI.

Solution Les dessins de la figure 3-44 b).

Symboles IEEE/ANSI pour des CI complexes Il n'y aurait pas d'avantage sensible à utiliser ces nouveaux symboles si on devait se limiter à représenter seulement des portes logiques élémentaires. Par contre, pour ce qui est des dispositifs logiques plus complexes, les nouveaux symboles normalisés et leur *notation fonctionnelle* permettent de spécifier complètement le fonctionnement logique du dispositif. Ceci rend presque inutile la consultation de la documentation du fabriquant afin de découvrir le rôle d'un CI logique dans un circuit. Nous illustrerons cet avantage au moyen d'exemples dans les chapitres à venir lorsque nous aborderons l'étude de circuits logiques plus complexes.

Tout au long de ce livre, nous continuerons d'employer les symboles logiques traditionnels dans les schémas de circuit. Et cela pour deux bonnes raisons: 1) les schémas des circuits logiques de la majorité de l'équipement numérique reposent encore sur la symbologie traditionnelle; et 2) il faudra compter plusieurs années encore avant que les milliers d'ingénieurs et de techniciens du domaine se convertissent à la nouvelle norme. Nous appliquerons cependant les nouveaux symboles IEEE/ANSI à la plupart des nouveaux circuits logiques que nous décrirons dans ce livre. De cette façon, vous deviendrez familiers avec les divers aspects de la notation fonctionnelle, le principal avantage introduit par la nouvelle norme.

QUESTIONS DE RÉVISION

1. Dites quel est le principal avantage des nouveaux symboles IEEE/ANSI.
2. Dessinez toutes les portes logiques de base en utilisant d'abord leurs symboles traditionnels, puis leurs symboles normalisés IEEE.
3. Même question que 2 ci-dessus mais pour chaque porte synonyme.

PROBLÈMES

SECTION 3-3

3-1 Tracez la forme d'onde de sortie du circuit de la figure 3-45.

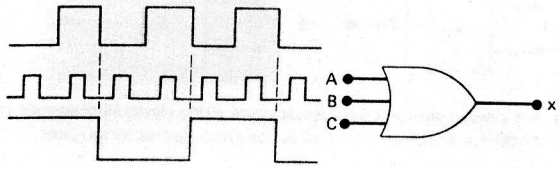


FIGURE 3-45

3-2 Supposez que l'entrée *A* de la figure 3-45 est court-circuitée à la masse par inadvertance (c.-à-d. $A = 0$). Tracez la forme d'onde de sortie résultante.

3-3 Supposez que l'entrée *A* de la figure 3-45 est court-circuitée à l'alimentation + 5 V par inadvertance (c.-à-d. $A = 1$). Tracez la forme d'onde de sortie résultante.

3-4 Lisez les énoncés suivants concernant une porte OU. Au premier coup d'oeil, ils vous sembleront valides, mais après quelques moments de réflexion, vous réaliserez qu'il n'y en a aucun qui soit *toujours* vrai. Faites la preuve de ceci en trouvant pour chaque énoncé un exemple qui le contredit.

a) Si la forme d'onde de sortie d'une porte OU est identique à la forme d'onde d'une de ses entrées, il faut conclure que l'autre entrée est toujours à 0.

b) Si la sortie d'une porte OU est toujours à 1, il faut conclure qu'une de ses entrées est aussi toujours à 1.

3-5 Combien d'ensembles différents de conditions d'entrée génèrent une sortie au niveau HAUT dans une porte OU à cinq entrées?

SECTION 3-4

3-6 Substituez à la porte OU de la figure 3-45 une porte ET.

a) Tracez la forme d'onde de sortie.

b) Tracez la forme d'onde de sortie si l'entrée *A* est court-circuitée à la masse.

c) Tracez la forme d'onde de sortie si l'entrée *A* est court-circuitée à l'alimentation.

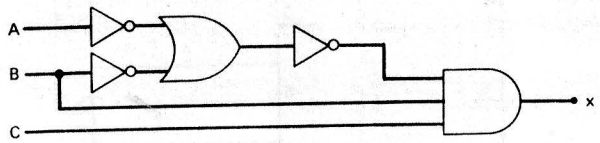
3-7 Reportez-vous à la figure 3-4. Modifiez ce circuit pour que l'alarme ne soit déclenchée que si la tempérture et la pression excèdent en même temps les valeurs de consigne maximales.

3-8 Substituez une porte OU à la porte ET dans la figure 3-6 et tracez la forme d'onde de sortie.

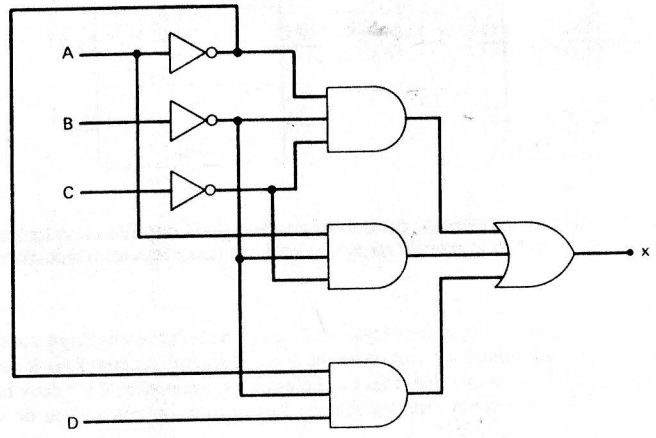
SECTIONS 3-5 à 3-7

3-9 Ajoutez un INVERSEUR à la sortie de la porte OU de la figure 3-45. Tracez la forme d'onde de la sortie de l'inverseur.

3-10 Écrivez l'expression booléenne de *x* de la figure 3-46 a). Trouvez la valeur de *x* pour toutes les conditions d'entrée possibles et dressez-en la liste dans une table de vérité.



a)



b)

FIGURE 3-46

3-11 Même question que celle du problème 3-10 mais pour le circuit de la figure 3-46 b).

3-12 Intervertissez les portes OU et ET de la figure 3-15 b) (c.-à-d. changer les portes OU par des portes ET et vice versa). Maintenant établissez l'expression pour la sortie.

3-13 Dressez la table de vérité complète du circuit de la figure 3-16 en trouvant le niveau logique de la sortie de chacune des portes pour les 16 combinaisons possibles des niveaux d'entrée.

SECTION 3-8

3-14 Pour chacune des expressions suivantes, construisez le circuit logique correspondant en recourant à des portes OU, des portes ET et des INVERSEURS.

a) $x = \overline{AB(C + D)}$

b) $z = (A + B + \overline{C}D\overline{E}) + \overline{B}C\overline{D}$

c) $y = (\overline{M + N} + \overline{P}Q)$

d) $x = \overline{W + P\overline{Q}}$

e) $z = MN(P + \overline{N})$

SECTION 3-9

3-15 Appliquez les formes d'ondes de la figure 3-47 aux trois entrées d'une porte NI et tracez la forme d'onde de sortie.

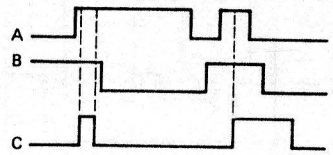


FIGURE 3-47

3-16 Même question que celle du problème 3-15 mais cette fois-ci en gardant l'entrée C toujours à 0. Ensuite en gardant C toujours égal à 1.

3-17 Même question que celle du problème 3-15 mais cette fois-ci pour une porte NON-ET.

3-18 Même question que celle du problème 3-16 mais cette fois-ci pour une porte NON-ET.

3-19 Écrivez l'expression de sortie du circuit de la figure 3-48.

3-20 Dressez la table de vérité complète du circuit de la figure 3-48.

3-21 Modifiez les circuits réalisés au problème 3-14 pour employer, lorsque cela s'impose, des portes NON-ET et des portes NI.

SECTION 3-10

3-22 CONTRÔLE

Complétez les expressions suivantes:

a) $A + 1 =$
b) $A \cdot A =$
c) $B \cdot \overline{B} =$
d) $C + C =$
e) $x \cdot 0 =$
f) $D \cdot 1 =$
g) $D + 0 =$
h) $C + \overline{C} =$
i) $G + GF =$
j) $y + \overline{w}y =$

3-23 Démontrez le théorème (15) en étudiant tous les cas possibles.

3-24 a) Simplifiez l'expression suivante en vous référant aux théorèmes (13b), (3) et (4).

$$x = (M + N)(\overline{M} + P)(\overline{N} + \overline{P})$$

b) Simplifiez l'expression suivante en vous référant aux théorèmes (13a), (8) et (6).

$$z = \overline{A}B\overline{C} + AB\overline{C} + B\overline{C}D$$

SECTIONS 3-11 et 3-12

3-25 Démontrez les théorèmes de De Morgan en étudiant tous les cas possibles.

3-26 Simplifiez chacune des expressions suivantes en utilisant les théorèmes de De Morgan.

a) $\overline{\overline{A}B\overline{C}}$
b) $\overline{\overline{A} + \overline{B}C}$
c) $\overline{AB\overline{C}D}$
d) $\overline{\overline{A(B + \overline{C})D}}$
e) $\overline{(M + \overline{N})(\overline{M} + N)}$
f) $\overline{\overline{A}\overline{B}CD}$

3-27 Recourez aux théorèmes de De Morgan pour simplifier l'expression de sortie de la figure 3-48.

3-28 Transformez le circuit de la figure 3.46 b) pour qu'il n'y ait que des portes NON-ET. Ensuite, écrivez l'expression de sortie de ce nouveau circuit, simplifiez-la avec les théorèmes de De Morgan, puis comparez-la à l'expression du circuit original.

3-29 Transformez le circuit de la figure 3-46 a) pour qu'il n'y ait que des portes NI. Ensuite, écrivez l'expression de sortie de ce nouveau circuit, simplifiez-la avec les théorèmes de De Morgan, puis comparez-la à l'expression du circuit original.

3-30 Montrez qu'il est possible de construire une porte NON-ET à deux entrées en utilisant des portes NI à deux entrées.

3-31 Montrez qu'il est possible de construire une porte NI à deux entrées en utilisant des portes NON-ET à deux entrées.

SECTIONS 3-13 et 3-14

3-32 Dessinez les représentations normales de chacune des portes logiques élémentaires. Ensuite, dessinez les représentations synonymes.

3-33 À chacun des énoncés suivants, faites correspondre un symbole de porte logique et nommez-en le type.

a) Un niveau HAUT est présent à la sortie seulement quand les trois entrées sont au niveau BAS.
b) Un niveau BAS est présent à la sortie seulement quand une des quatre entrées est au niveau BAS.
c) Un niveau BAS est présent à la sortie seulement quand toutes les huit entrées sont au niveau HAUT.

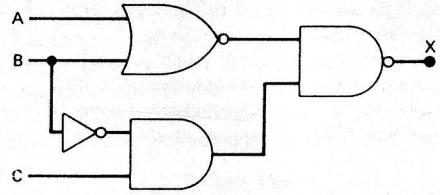


FIGURE 3-48

3-34 La sortie de la figure 3-48 doit allumer une lampe témoin quand elle passe au niveau BAS.

a) Modifiez le schéma de circuit afin qu'il dépeigne mieux le fonctionnement du circuit.
b) À partir de ce nouveau schéma, déterminez les conditions d'entrée qui vont valider (activer) la sortie. Procédez à rebours à partir de la sortie en utilisant les informations fournies par les symboles logiques comme dans les exemples 3-22 et 3-23. Comparez vos résultats avec la table de vérité obtenue dans le problème 3-20.

3-35 a) Déterminez les conditions d'entrée qui activent la sortie Z de la figure 3-37 b). Procédez à rebours en partant de la sortie comme dans les exemples 3-22 et 3-23.

b) Supposez que Z déclenche l'alarme quand il devient au niveau BAS. Changez le schéma de circuit afin qu'il évoque ce fait, puis révisez le schéma pour déterminer les conditions d'entrée qui déclenchent l'alarme.

3-36 Modifiez le circuit de la figure 3-40 de manière qu'on ait besoin de $A_1 = 0$ pour obtenir $CASS = 1$ plutôt que $A_1 = 1$.

3-37 Trouvez les conditions d'entrée qui font en sorte que la sortie de la figure 3-49 passe à son niveau vrai.

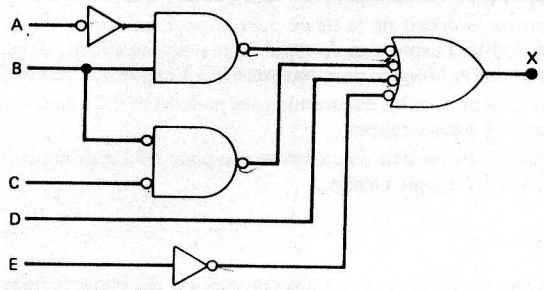


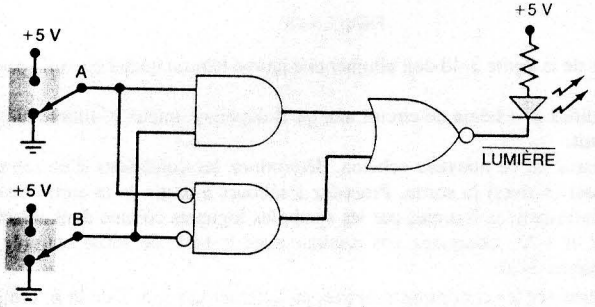
FIGURE 3-49

3-38 À l'aide des résultats trouvés au problème 3-37, construisez la table de vérité complète du circuit de la figure 3-49.

3-39 Quel est l'état valide de la sortie sur la figure 3-49? Celui de la sortie sur la figure 3-36 c)?

3-40 La figure 3-50 nous montre l'utilisation des portes logiques pour simuler un interrupteur bidirectionnel comme ceux qui allument et éteignent la lumière à partir de deux commutateurs. Dans le cas présent, la lumière est émise par une DEL qui conduit quand la sortie de la porte NI est BASSE. Remarquez le nom donné à cette sortie, soit *LUMIÈRE* pour indiquer qu'elle est vraie au niveau BAS. Déterminez les conditions d'entrée qui font briller la DEL. Ensuite, vérifiez, au moyen des commutateurs A et B, que ce circuit reproduit le fonctionnement d'un interrupteur bidirectionnel. Au chapitre 4, vous apprendrez comment concevoir des circuits comme ce dernier qui établissent un rapport entre les entrées et les sorties.

FIGURE 3-50



SECTION 3-15

3-41 Redessinez le circuit de la figure 3-49 en utilisant la symbologie IEEE/ANSI.

3-42 Trouvez l'expression booléenne de la sortie Z de la figure 3-51.

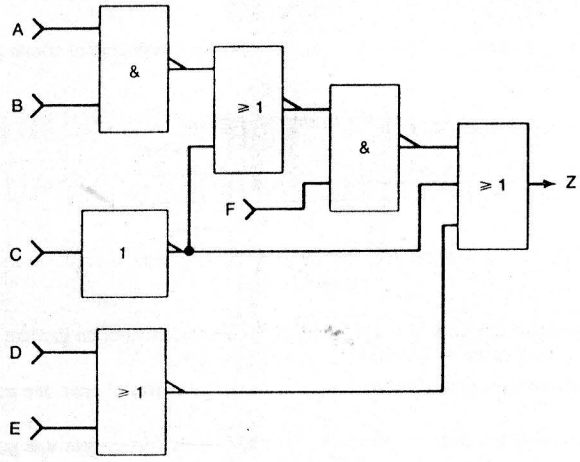


FIGURE 3-51

***3-43** La sortie du circuit de la figure 3-51 est supposée vraie au niveau BAS. Redessinez ce circuit afin de mieux évoquer son fonctionnement.

***3-44** En vous servant du schéma redessiné au problème 3-51, répondez aux questions suivantes:

a) Déterminez les différentes conditions d'entrée qui donnent lieu à un état de sortie vrai au niveau BAS. Accomplissez ceci en vous servant essentiellement du schéma du circuit sans trouver l'expression Z et sans produire la table de vérité. Vous devriez trouver:

A	B	C	D	E	F
1	1	1	1	1	1
1	1	1	1	0	1
1	1	1	0	1	1

b) Vérifiez que l'expression simplifiée pour la sortie Z est:

$$Z = \overline{ABCF(D + E)}$$

c) Substituez chacune des solutions données en a) dans l'expression déterminée en b) et vérifiez que $Z = 0$ dans tous les cas.

APPLICATION AU MICRO-ORDINATEUR

3-45 Reportez-vous à l'exemple 3-23. Les entrées A_7 à A_0 sont les bits de l'adresse fournie à ce circuit par les broches de sortie de la puce microprocesseur d'un micro-ordinateur. Ce code d'adresse de 8 bits A_7 à A_0 sélectionne le dispositif que le microprocesseur veut activer. Dans l'exemple 3-23, le code envoyé pour activer l'enregistreur à cassette est A_7 à $A_0 = 11111110_2 = FE_{16}$.

Modifiez ce circuit pour que le microprocesseur ait à envoyer le code $4A_{16}$ pour actionner l'enregistreur.

* Une ampoule à gauche annonce un problème particulièrement ardu.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 3-3

1. Toutes les entrées sont BASSES.
2. $x = A + B + C + D + E + F$
3. Niveau HAUT constant.

SECTION 3-4

1. Les cinq entrées = 1.
2. Une entrée BASSE garde la sortie BASSE.
3. Faux. Voyez la table de vérité de chaque porte.

SECTION 3-5

1. La sortie du second INVERSEUR est identique à l'entrée A. **2.** y est au niveau BAS seulement quand $A = B = 1$.

SECTION 3-6

1. $x = \overline{A} + B + C + \overline{AD}$

SECTION 3-7

1. $x = 1$ **2.** $x = 1$

SECTION 3-8

1. Voyez figure 3-15 a).

SECTION 3-9

1. Toutes les entrées sont BASSES. **2.** Porte ET. **3.** $x = \overline{A + B + \overline{CD}}$

SECTION 3-10

1. $y = A\overline{C}$ $y = \overline{ABD}$

SECTION 3-11

1. $z = \overline{AB} + C$ **2.** $y = (\overline{R} + S + \overline{T})Q$
3. Même que sur la figure 3-28 sauf qu'une porte NON-ET remplace la porte OU.

SECTION 3-12

1. Trois. **2.** Le circuit NI est plus efficace puisqu'on peut le réaliser avec un seul CI 7402.

SECTION 3-13

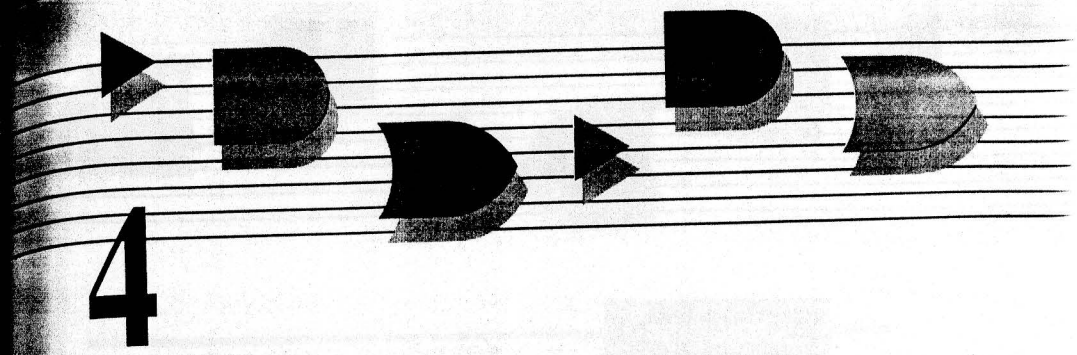
1. La sortie passe à BAS quand au moins une entrée est HAUTE. **2.** La sortie passe à HAUT quand toutes les entrées sont BASSES. **3.** La sortie devient BASSE quand une entrée est BASSE. **4.** La sortie passe à HAUT seulement quand toutes les entrées sont HAUTES.

SECTION 3-14

1. Z devient HAUT quand $A = B = 0$ et $C = D = 1$. **2.** Z devient BAS quand $A = B = 0$, $E = 1$ et C ou D, ou les deux, sont à 0. **3.** Deux. **4.** Deux. **5.** BAS.
6. $A = B = 0$, $C = D = 1$.

SECTION 3-15

1. Les symboles IEEE et leur notation fonctionnelle spécifient complètement le fonctionnement du dispositif logique. **2.** Voyez la figure 3-41. **3.** Voyez la figure 3-44.

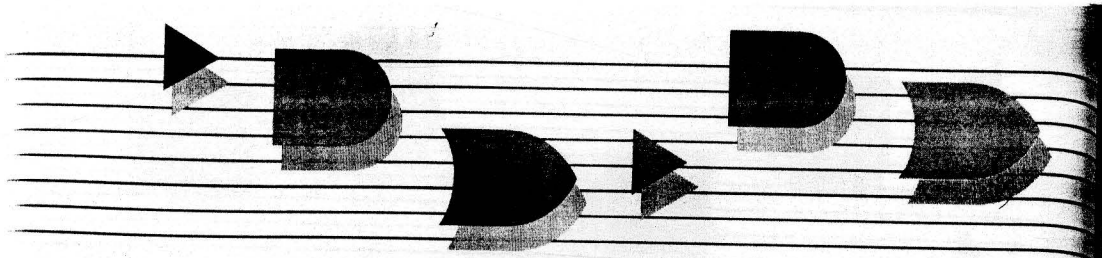


4

CIRCUITS LOGIQUES COMBINATOIRES

Contenu

4-1 Somme de produits
4-2 Simplification des circuits logiques
4-3 Simplification algébrique
4-4 Conception de circuits logiques combinatoires
4-5 La méthode des diagrammes de Karnaugh
4-6 Circuits OU exclusif et NI exclusif
4-7 Générateur et contrôleur de parité
4-8 Circuits de blocage
4-9 Caractéristiques de base des circuits numériques
4-10 Dépannage des systèmes numériques
4-11 Défaillances internes des CI numériques
4-12 Défaillances externes
4-13 Étude d'un cas de dépannage



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Convertir une expression logique en une somme de produits.
- Mener à bien les étapes nécessaires pour aboutir à une somme de produits en vue de concevoir un circuit logique combinatoire minimisé.
- Utiliser un diagramme de Karnaugh comme instrument pour simplifier et concevoir un circuit logique.
- Concevoir des circuits logiques en recourant ou non à une table de vérité.
- Reconnaître et de comprendre les circuits intégrés numériques.
- Citer les caractéristiques de base des circuits intégrés numériques.
- Expliquer les différences de fonctionnement intrinsèques entre un élément TTL et un élément CMOS.
- Mettre en application les règles de dépannage élémentaires des systèmes numériques.
- Déduire à partir des mesures effectuées les causes de mauvais fonctionnement des circuits logiques combinatoires.

Introduction

Au chapitre 3, nous nous sommes penchés sur l'étude de l'ensemble des portes logiques élémentaires et avons réussi, au moyen de l'algèbre booléenne, à décrire et à analyser des circuits matérialisés par des combinaisons de portes logiques. On peut qualifier ces circuits logiques de *combinatoires* du fait qu'à tout moment le niveau logique recueilli en sortie ne dépend que de la combinaison des niveaux logiques appliqués aux entrées. Un circuit combinatoire ne possède aucun mécanisme de rétention (mémoire); par conséquent, sa sortie réagit seulement aux signaux présents sur ses entrées.

Dans ce chapitre, nous poursuivons notre étude des circuits combinatoires. D'abord, nous poussons plus avant la simplification (minimisation) des circuits. Pour cela, nous utiliserons deux méthodes: les théorèmes de l'algèbre booléenne et une technique *graphique*. En outre, nous verrons quelles techniques employer pour concevoir des circuits logiques répondant à certaines conditions. Il n'entre pas dans le cadre de ce livre de traiter en détail l'étude des circuits logiques; cependant, les méthodes abordées ici devraient suffire pour les types de circuits qu'un technicien est appelé à concevoir.

Les dernières sections du présent chapitre sont consacrées au dépannage des circuits combinatoires. Cette initiation au dépannage vise à développer chez l'étudiant ses aptitudes d'analyse qui lui permettront de réussir les travaux de dépannages. Dans le but de rendre la matière la plus pratique possible, nous exposons, en premier lieu, les caractéristiques de base des CI des familles logiques TTL et CMOS et nous poursuivons avec une description des types de défaillances les plus courantes dans les circuits intégrés numériques.

4-1 SOMME DE PRODUITS

Les méthodes de simplification et de conception des circuits logiques que nous étudierons exigent que l'on exprime les équations logiques sous la forme d'une *somme de produits*, dont voici quelques exemples:

1. $ABC + \overline{A}B\overline{C}$
2. $AB + \overline{A}B\overline{C} + \overline{C}\overline{D} + D$
3. $\overline{A}B + C\overline{D} + EF + GK + H\overline{L}$

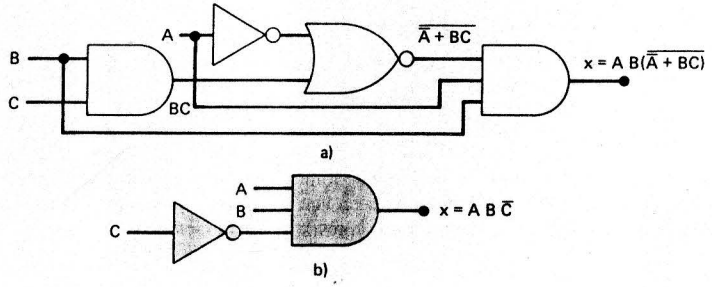
Chacune de ces trois expressions d'une somme de produits est formée d'au moins deux résultats d'un produit logique (terme ET) mis en relation OU (additionnés logiquement). Chaque terme ET comprend une ou plusieurs variables exprimées sous sa forme normale ou sa forme complémentée. Par exemple, dans la somme de produits $ABC + \overline{A}B\overline{C}$, le premier produit logique est constitué des variables A, B et C non complémentées, tandis que le second produit comprend les variables A et C complémentées. Notez que dans une somme de produits, le signe de complémentation *ne peut pas* surmonter plus d'une variable d'un terme (par ex. on ne peut pas avoir $\overline{ABC}$ ou $\overline{RST}$).

4-2 SIMPLIFICATION DES CIRCUITS LOGIQUES

Dès qu'on dispose de l'expression d'un circuit logique, il peut être possible de la minimiser pour obtenir une équation comptant moins de termes ou moins de variables par terme. Cette nouvelle équation peut alors servir de modèle pour construire un circuit entièrement équivalent au circuit original mais qui requiert moins de portes et de raccordements.

À titre d'illustration, considérons le circuit de la figure 4-1 a) que l'on a minimisé pour obtenir le circuit de la figure 4-1 b). Étant donné que ces deux circuits produisent les mêmes décisions logiques, il va sans dire que le circuit le plus simple est préférable puisqu'il compte moins de portes; il est aussi moins encombrant et moins coûteux à produire. De

FIGURE 4-1 Il est souvent possible de simplifier un circuit comme en a) ci-dessous pour obtenir une réalisation plus efficace comme en b).



plus, c'est un circuit plus fiable parce qu'il comprend moins d'interconnexions pouvant donner lieu éventuellement à des pannes.

Dans les sections suivantes, nous verrons deux façons différentes de simplifier des circuits logiques. Une première façon est fondée sur l'application des théorèmes de l'algèbre booléenne; cette façon, comme nous le verrons, dépend énormément de l'instinct et de l'expérience. L'autre façon (les diagrammes de Karnaugh), au contraire, suit une démarche systématique, semblable à une recette de cuisine. Certains professeurs choisiront de ne pas parcourir cette dernière méthode, parce que selon eux, elle est strictement mécanique et n'apporte rien à la compréhension de l'algèbre booléenne. C'est un choix valable qui n'affecte aucunement la continuité ou la clarté de la matière présentée par la suite.

4-3 SIMPLIFICATION ALGÉBRIQUE

Les théorèmes de l'algèbre booléenne étudiés au chapitre 3 peuvent nous être d'un grand secours pour simplifier l'expression d'un circuit logique. Malheureusement, il n'est pas toujours facile de savoir quels théorèmes il faut invoquer pour obtenir le résultat minimal. D'ailleurs, rien ne nous dit que l'expression simplifiée est la forme minimale et qu'il n'y a pas d'autres simplifications possibles. Pour toutes ces raisons, la simplification algébrique est souvent un processus d'approximations successives. L'expérience aidant, une personne peut exceller à trouver d'assez bons résultats.

Les exemples qui suivent montrent diverses manières d'utiliser les théorèmes de l'algèbre de Boole en vue de simplifier une expression. Nous tenons à vous faire remarquer que dans tous ces exemples on retrouve toujours deux étapes essentielles:

1. La transformation par applications successives des théorèmes de De Morgan et par multiplication de termes de l'expression pour obtenir une somme de produits.
2. Ensuite, la vérification de chaque produit pour trouver les variables communes, puis la mise en facteur de ces dernières, le cas échéant. Quand une simplification est possible, la mise en facteur permet d'éliminer un ou plusieurs termes.

EXEMPLE 4-1

Simplifiez le circuit logique illustré à la figure 4-2 a).

Solution La première étape consiste à établir l'expression de la sortie. Cette expression est:

$$z = ABC + A\overline{B} \cdot (\overline{\overline{A}\,\overline{C}})$$

Quand on a l'expression, il est souvent judicieux de décomposer, au moyen des théorèmes de De Morgan, les longs signes de complémentation et ensuite de multiplier tous les termes.

$$\begin{aligned} z &= ABC + A\overline{B}(\overline{\overline{A}} + \overline{\overline{C}}) && \text{[théorème 17]} \\ &= ABC + A\overline{B}(A + C) && \text{[annulation de la double complémentation]} \\ &= ABC + A\overline{B}A + A\overline{B}C && \text{[multiplication]} \\ &= ABC + A\overline{B} + A\overline{B}C && [A \cdot A = A] \end{aligned}$$

Maintenant que nous avons en main la somme de produits recherchée, voyons les variables communes dans les divers termes en vue de mettre en facteur. Le premier et le troisième terme ont AC en commun; d'où, après la mise en facteur:

$$z = AC(B + \overline{B}) + A\overline{B}$$

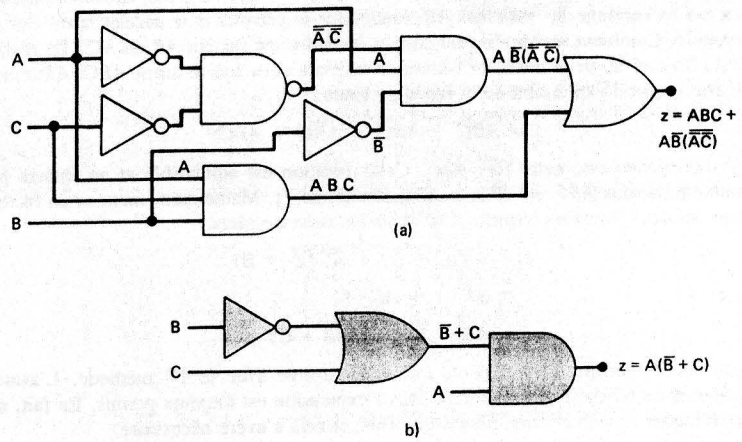


FIGURE 4-2 Exemple 4-1.

Comme $B + \overline{B} = 1$, alors

$$\begin{aligned} z &= AC(1) + A\overline{B} \\ &= AC + A\overline{B} \end{aligned}$$

En mettant en facteur A, il vient:

$$z = A(C + \overline{B})$$

Ce résultat est la forme minimale de l'expression. La matérialisation du circuit correspondant est donnée à la figure 4-2 b) qui est, il va sans dire, nettement plus simple que le circuit original que l'on voit en a).

EXEMPLE 4-2

Simplifiez l'expression $z = ABC + AB\overline{C} + A\overline{B}C$.

Solution Nous allons trouver le même résultat au moyen de deux méthodes différentes.

1re méthode: Les deux premiers termes de l'expression ont les variables AB en commun. D'où

$$\begin{aligned} z &= AB(C + \overline{C}) + A\overline{B}C \\ &= AB(1) + A\overline{B}C \\ &= AB + A\overline{B}C \end{aligned}$$

Il est possible ici de mettre en facteur A, de sorte que

$$z = A(B + \overline{B}C)$$

Ce qui en raison du théorème (15) devient:

$$z = A(B + C)$$

2e méthode: L'expression originale est $z = ABC + AB\overline{C} + A\overline{B}C$. Les deux premiers termes ont en commun les variables AB, tandis que le premier et le dernier terme ont AC en commun. Comment savoir s'il faut choisir de mettre en facteur AB ou AC? En réalité, on peut effectuer les deux mises en facteur en utilisant *deux* fois le terme ABC. Autrement dit, il faut récrire l'expression de la façon suivante:

$$z = ABC + AB\overline{C} + A\overline{B}C + ABC$$

c'est-à-dire ajouter une autre fois ABC. Cette addition est admissible et ne change pas l'expression puisque $ABC + ABC = ABC$ [théorème 7]. Maintenant, mettons en facteur AB dans les deux premiers termes et AC dans les deux derniers.

$$\begin{aligned} z &= AB(C + \overline{C}) + AC(\overline{B} + B) \\ &= AB \cdot 1 + AC \cdot 1 \\ &= AB + AC = A(B + C) \end{aligned}$$

Évidemment, le résultat est le même que celui trouvé avec la 1re méthode. L'astuce d'additionner un terme se trouvant déjà dans l'expression est toujours permis. En fait, on peut additionner le même terme plus d'une fois, si cela s'avère nécessaire.

EXEMPLE 4-3

Simplifiez $z = \overline{A}C(\overline{\overline{A}BD}) + \overline{A}B\overline{C}\overline{D} + A\overline{B}C$.

Solution Premièrement, appliquez le théorème de De Morgan au premier terme.

$$z = \overline{A}C(A + \overline{B} + \overline{D}) + \overline{A}B\overline{C}\overline{D} + A\overline{B}C \qquad (1^{re}\text{ étape})$$

Ensuite multipliez

$$z = \overline{A}CA + \overline{A}C\overline{B} + \overline{A}C\overline{D} + \overline{A}B\overline{C}\overline{D} + A\overline{B}C \qquad (2)$$

Comme $\overline{A} \cdot A = 0$, on peut éliminer le premier terme.

$$z = \overline{A}\overline{B}C + \overline{A}C\overline{D} + \overline{A}B\overline{C}\overline{D} + A\overline{B}C \qquad (3)$$

Mettez en facteur $\overline{B}C$ dans le premier et le dernier terme, et $\overline{A}\overline{D}$ dans le second et le troisième terme.

$$z = \overline{B}C(\overline{A} + A) + \overline{A}\overline{D}(C + B\overline{C}) \qquad (4)$$

Puisque $\overline{A} + A = 1$ et $C + B\overline{C} = C + B$ [théorème 15], il vient:

$$z = \overline{B}C + \overline{A}\overline{D}(C + B) \qquad (5)$$

Le même résultat pourrait être obtenu en mettant en facteur différemment. Par exemple, on aurait pu mettre en facteur C à l'étape 3, ce qui donne:

$$z = C(\overline{A}\overline{B} + \overline{A}\overline{D} + A\overline{B}) + \overline{A}B\overline{C}\overline{D}$$

L'expression entre parenthèses contient un terme commun que l'on peut mettre en facteur:

$$z = C(\overline{B}[\overline{A} + A] + \overline{A}\overline{D}) + \overline{A}B\overline{C}\overline{D}$$

Comme $\overline{A} + A = 1$ cette dernière expression devient:

$$z = C(\overline{B} + \overline{A}\overline{D}) + \overline{A}B\overline{C}\overline{D}$$

En multipliant, il vient:

$$z = \overline{B}C + \overline{A}C\overline{D} + \overline{A}B\overline{C}\overline{D}$$

À présent la mise en facteur de $\overline{A}\overline{D}$ dans le deuxième et le troisième termes produit:

$$z = \overline{B}C + \overline{A}\overline{D}(C + B\overline{C})$$

D'après le théorème (15), l'expression entre parenthèses vaut $B + C$. Ce qui en définitive donne:

$$z = \overline{B}C + \overline{A}\overline{D}(B + C)$$

Soit le même résultat que celui en (5), obtenu au moyen de plus d'étapes. Cela démontre pourquoi vous devez rechercher le groupe de facteurs communs le plus grand, puisque c'est généralement avec ce groupe qu'on arrive le plus vite à l'expression finale.

EXEMPLE 4-4

Simplifiez l'expression $x = (\overline{A} + B)(A + B + D)\overline{D}$.

Solution En multipliant tous les termes, on transforme cette expression en somme de produits, ce qui donne:

$$x = \overline{A}A\overline{D} + \overline{A}B\overline{D} + \overline{A}D\overline{D} + BA\overline{D} + BB\overline{D} + BD\overline{D}$$

Éliminons le premier terme puisque $\overline{A}A = 0$, le troisième et le sixième terme, pour la même raison, puisque $D\overline{D} = 0$. Enfin, puisque $BB = B$, simplifions le cinquième terme. Tout ceci donne:

$$x = \overline{A}B\overline{D} + AB\overline{D} + B\overline{D}$$

Mettons en facteur $B\overline{D}$ se trouvant dans chacun de ces termes,

$$x = B\overline{D}(\overline{A} + A + 1)$$

Comme il est évident que la somme entre parenthèses égale 1, il vient en dernier lieu:

$$x = B\overline{D}$$

EXEMPLE 4-5

Simplifiez le circuit de la figure 4-3 a).

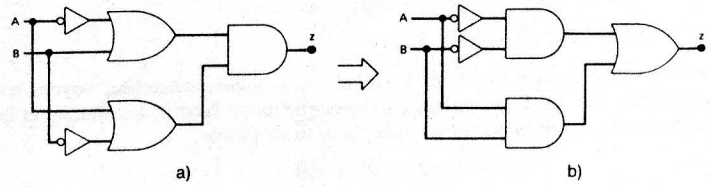
Solution L'expression de la sortie z de ce circuit est:

$$z = (\overline{A} + B)(A + \overline{B})$$

En multipliant, on obtient la somme de produits suivante:

$$z = \overline{A}A + \overline{A}\overline{B} + BA + B\overline{B}$$

FIGURE 4-3 Exemple 4-5.



a) b)

En éliminant les termes $\overline{A}A = 0$ et $B\overline{B} = 0$, il reste

$$z = \overline{A}\overline{B} + AB$$

Le circuit qui matérialise cette expression est montré à la figure 4-3 b). Si on compare ce dernier circuit avec l'original, on voit qu'ils ont tous les deux le même nombre de portes et de raccordements. La simplification dans ce cas-ci a produit un circuit équivalent mais pas plus simple que l'original.

EXEMPLE 4-6

Simplifiez $x = A\overline{B}C + \overline{A}BD + \overline{C}\overline{D}$.

Solution Vous pouvez toujours essayer, mais il n'y a pas moyen d'obtenir une expression plus simple que celle-ci.

QUESTIONS DE RÉVISION

1. Parmi les expressions suivantes, indiquez celles qui ne sont pas exprimées sous la forme d'une somme de produits. a) $RS\overline{T} + \overline{R}S\overline{T} + \overline{T}$; b) $A\overline{D}\overline{C} + \overline{A}DC$; c) $MN\overline{P} + (M + \overline{N})P$; d) $AB + \overline{A}B\overline{C} + AB\overline{C}D$.
2. Simplifiez le circuit de la figure 4-1 a) pour aboutir au circuit de la figure 4-1 b).
3. Remplacez toutes les portes ET de la figure 4-1 a) par des portes NON-ET. Déterminez la nouvelle expression pour x et simplifiez-la.

4-4 CONCEPTION DE CIRCUITS LOGIQUES COMBINATOIRES

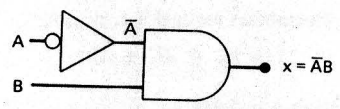
Quand on indique le niveau logique recherché pour toutes les conditions d'entrée possibles, il est facile de mettre les résultats sous la forme d'une table de vérité. Ensuite, on peut dériver l'expression booléenne du circuit à partir de cette table de vérité. Par exemple, soit la figure 4-4 a) où l'on voit une table de vérité décrivant le fonctionnement d'un circuit à deux entrées, A et B, et à une sortie, x. Cette table montre que la sortie x est à 1 *seulement* quand $A = 0$ et $B = 1$. Aussi ne reste-t-il qu'à trouver le circuit logique qui réalise la fonction désirée. Une solution possible est naturellement le circuit illustré à la figure 4-4 b), où une porte ET effectue le produit logique des entrées $\overline{A}$ et B, de sorte que $x = \overline{A} \cdot B$. On voit immédiatement que $x = 1$ *seulement si* $\overline{A} = 1$ (c.-à-d. $A = 0$) et $B = 1$. Pour toutes les autres valeurs de A et B, la sortie x est nulle.

On peut procéder à l'étude d'autres conditions d'entrée en suivant une approche similaire. Par exemple, si x est au niveau HAUT seulement quand $A = 1$ et $B = 0$, le circuit résultant sera une porte ET dont les entrées seront A et $\overline{B}$. Au total, pour n'importe

FIGURE 4-4

A	B	x
0	0	0
0	1	1
1	0	0
1	1	0

a)



b)

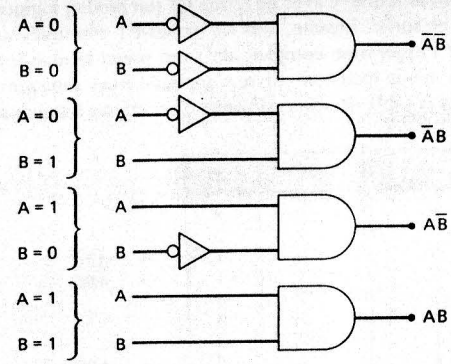


FIGURE 4-5 Les portes ET qui donnent un niveau haut pour chacune des quatre conditions d'entrée possibles.

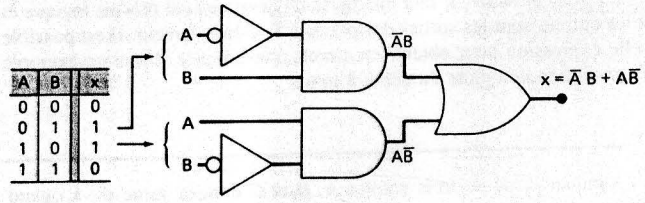
laquelle des quatres conditions d'entrée possibles, on peut obtenir une valeur 1 en sortie en utilisant une porte ET et en appliquant les bonnes entrées. D'ailleurs, on peut voir ces quatre circuits à la figure 4-5. Chacune des portes ET illustrée génère un 1 *uniquement* pour la condition d'entrée donnée et un 0 pour toutes les autres combinaisons. Il convient de remarquer que les entrées de la porte ET sont complémentées ou non selon les valeurs des variables qui sont spécifiées comme conditions d'entrée. Ainsi quand une variable doit être 0 dans une condition donnée, cette variable est complémentée avant d'être introduite dans la porte ET.

Voyons le cas exposé par la figure 4-6 a) où une table de vérité montre que la sortie x est 1 pour deux cas possibles, soit $A = 0$, $B = 1$ et $A = 1$, $B = 0$. Comment s'y prendre pour trouver le circuit correspondant à cela? Nous savons que le produit logique $\overline{A} \cdot B$ donne 1 seulement quand la condition $A = 0$, $B = 1$ est réalisée, et que le produit logique $A \cdot \overline{B}$ donne 1 quand l'autre condition $A = 1$, $B = 0$ est réalisée. Comme x doit prendre la valeur 1 pour *l'une ou l'autre* de ces conditions, on en déduit immédiatement qu'il suffit d'additionner logiquement ces deux termes ET pour obtenir la sortie x recherchée. Le circuit correspondant est illustré à la figure 4-6 b) dont l'expression de sortie est $x = \overline{A} \cdot B + A \cdot \overline{B}$.

FIGURE 4-6 Chaque ensemble de conditions d'entrée qui produit une sortie HAUTE est matérialisé en une porte ET. Toutes les sorties des portes résultantes sont additionnées logiquement (OU) pour donner la réponse finale.

A	B	x
0	0	0
0	1	1
1	0	1
1	1	0

a)



b)

Dans cet exemple, on se retrouve avec un terme ET (un produit logique) pour chaque condition qui donne un 1 en sortie. Ensuite, tous les termes ET constitués sont additionnés logiquement pour obtenir l'expression complète de x qui prend le niveau haut quand un des termes ET est à 1. La même marche à suivre s'applique à des situations ayant plus de deux entrées. Considérons la table de vérité suivante d'un circuit à trois entrées:

A	B	C	x	
0	0	0	0	
0	0	1	0	
0	1	0	1	→ $\overline{A}B\overline{C}$
0	1	1	1	→ $\overline{A}BC$
1	0	0	0	
1	0	1	0	
1	1	0	0	
1	1	1	1	→ ABC

On peut voir qu'il y a trois cas qui produisent une valeur 1 pour la sortie x. Les termes ET pour chacun de ces cas figurent en regard de la table de vérité. Notez de nouveau que chaque fois qu'une variable doit être 0 dans une condition qui donnent 1, celle-ci est complémentée dans le produit logique correspondant. L'expression définitive pour x s'obtient en additionnant logiquement ces trois termes, ce qui donne:

$$x = \overline{A}B\overline{C} + \overline{A}BC + ABC$$

Il est possible de matérialiser cette expression en ayant trois portes ET à trois entrées dont les sorties alimentent une porte OU à trois entrées.

Dans tous les exemples qui précèdent, l'expression pour la sortie x découle de la table de vérité et se présente sous la forme d'une somme de produits. Voici une récapitulation de la procédure générale qui aboutit à l'expression de la sortie à partir d'une table de vérité:

1. Pour chaque cas de la table qui donne 1 en sortie, on écrit le produit logique (terme ET) qui lui correspond.
2. On doit retrouver toutes les variables d'entrée dans chaque terme ET soit sous forme directe soit sous forme complémentée. Dans un cas particulier, si une variable est 0, alors son symbole est complémenté dans le terme ET correspondant.
3. On additionne logiquement ensuite tous les produits logiques constitués, ce qui donne l'expression définitive de la sortie.

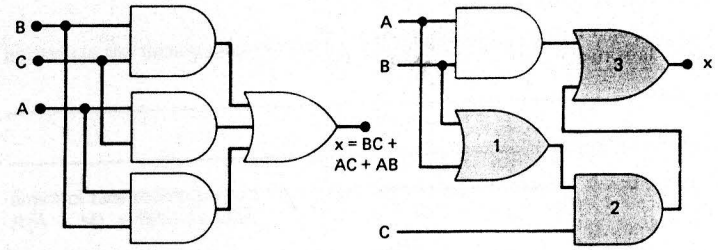
La conception complète Dès que l'expression de la sortie est établie sous la forme d'une somme de produits a partir de la table de vérité, il est facile de construire le circuit au moyen de portes ET, OU et NON. Il faut une porte ET pour chaque produit logique et une porte OU dont les entrées sont les sorties des portes ET. Généralement, il est possible de simplifier une telle expression pour obtenir un circuit plus efficace. Voici un exemple qui illustre bien la conception complète du début à la fin.

EXEMPLE 4-7

Concevez un circuit logique formé de trois entrées A, B et C dont la sortie est 1 quand une majorité des entrées sont à 1.

A	B	C	x	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\overline{A}BC$
1	0	0	0	
1	0	1	1	$A\overline{B}C$
1	1	0	1	$AB\overline{C}$
1	1	1	1	ABC

a)



b) c)

FIGURE 4-7 Exemple 4-7.

Solution La première étape consiste à préparer la table de vérité en se fondant sur l'énoncé du problème. Les huit combinaisons d'entrée possibles sont montrées à la figure 4-7 a). D'après l'énoncé du problème, la sortie x sera à 1 si au moins deux des entrées sont à 1. Dans tous les autres cas, la sortie sera 0. La deuxième étape se réduit à écrire les termes ET pour chaque cas donnant $x = 1$, ceux figurant en regard de la table. L'expression de la sortie est donc:

$$x = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$

Il est possible de simplifier cette expression de plusieurs manières, la plus rapide étant peut-être d'exploiter le fait que ABC a deux variables communes avec chacun des autres termes. On peut donc utiliser le terme ABC que l'on additionne deux autres fois dans l'expression (rappelons-nous que c'est un artifice admis en algèbre booléenne) pour effectuer trois mises en facteur. On a alors:

$$x = \overline{A}BC + ABC + A\overline{B}C + ABC + AB\overline{C} + ABC$$

En mettant en facteur les variables communes des paires appropriées, on obtient:

$$x = BC(\overline{A} + A) + AC(\overline{B} + B) + AB(\overline{C} + C)$$

Comme chaque terme entre parenthèses est égal à 1, il vient:

$$x = BC + AC + AB$$

Cette expression est matérialisée à la figure 4-7 b). Étant donné que cette expression est une somme de produits, il y a dans ce circuit un groupe de portes ET dont les sorties alimentent une seule porte OU.

Il est possible d'effectuer une autre mise en facteur dans cette expression; cependant, le résultat ne permet pas d'éliminer d'autres termes. Il permet toutefois de construire un circuit légèrement plus simple. Ainsi, mettons le C des deux premiers termes en facteur, ce qui donne

$$x = C(B + A) + AB$$

On peut voir le circuit correspondant à cette expression à la figure 4-7 c). Ce circuit contient le même nombre de portes que le circuit précédent, mais requiert une connexion de moins, car la dernière porte OU n'a que deux entrées. Dans la pratique, un concepteur pourrait, malgré cette connexion en moins, choisir le circuit de la figure 4-7 b) en prenant en considération d'autres motifs. Entre autres, il pourrait considérer que les signaux A et B doivent se propager à travers trois portes (1, 2 et 3) dans le circuit de la figure 4-7 c) avant de produire la sortie. C'est un inconvénient non négligeable dans un système numérique rapide.

EXEMPLE 4-8

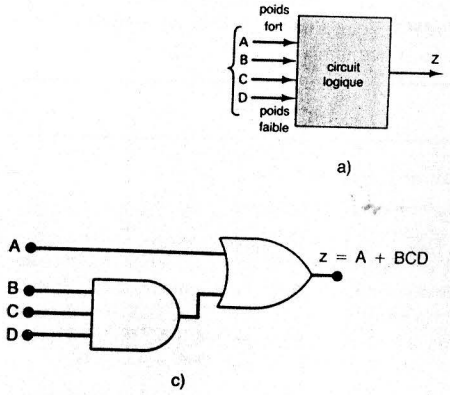
Reportez-vous à la figure 4-8 a) où quatre lignes logiques, appelées A, B, C et D, forment un nombre binaire dont le bit de poids le plus fort est A et le bit de poids le plus faible est D. Ces entrées binaires sont appliquées à un circuit logique qui donne en sortie un niveau HAUT quand le chiffre présent est supérieur à $0110_2 = 6_{10}$. Trouvez ce circuit.

Solution La table de vérité correspondante est illustrée à la figure 4-8 b). Pour chaque ligne de la table, nous donnons entre parenthèses le nombre décimal équivalent au nombre binaire représenté par la combinaison $ABCD$.

La sortie z prend la valeur 1 lorsque le nombre binaire est supérieur à 0110. Pour tous les autres cas, z est fixé à 0. De cette table de vérité, on tire la somme de produits suivante:

$$z = \bar{A}BCD + A\bar{B}\bar{C}\bar{D} + A\bar{B}C\bar{D} + A\bar{B}CD + AB\bar{C}\bar{D} + AB\bar{C}D + ABC\bar{D} + ABCD$$

FIGURE 4-8 Exemple 4.8.



	A	B	C	D	z	
(0)	0	0	0	0	0	
(1)	0	0	0	1	0	
(2)	0	0	1	0	0	
(3)	0	0	1	1	0	
(4)	0	1	0	0	0	
(5)	0	1	0	1	0	
(6)	0	1	1	0	0	
(7)	0	1	1	1	1	$\rightarrow \bar{A}BCD$
(8)	1	0	0	0	1	$\rightarrow A\bar{B}\bar{C}\bar{D}$
(9)	1	0	0	1	1	$\rightarrow A\bar{B}\bar{C}D$
(10)	1	0	1	0	1	$\rightarrow A\bar{B}C\bar{D}$
(11)	1	0	1	1	1	$\rightarrow A\bar{B}CD$
(12)	1	1	0	0	1	$\rightarrow AB\bar{C}\bar{D}$
(13)	1	1	0	1	1	$\rightarrow AB\bar{C}D$
(14)	1	1	1	0	1	$\rightarrow ABC\bar{D}$
(15)	1	1	1	1	1	$\rightarrow ABCD$

b)

La simplification de cette expression est une tâche colossale, mais que l'on peut mener à bien si on y apporte le soin nécessaire. La marche à suivre implique la mise en facteur des variables communes et l'élimination des termes de la forme $(A + \bar{A})$.

$$\begin{aligned} z &= \bar{A}BCD + A\bar{B}\bar{C}(\bar{D} + D) + A\bar{B}C(\bar{D} + D) + AB\bar{C}(\bar{D} + D) \\ &\quad + ABC(\bar{D} + D) \\ &= \bar{A}BCD + A\bar{B}\bar{C} + A\bar{B}C + AB\bar{C} + ABC \\ &= \bar{A}BCD + A\bar{B}(\bar{C} + C) + AB(\bar{C} + C) \\ &= \bar{A}BCD + A\bar{B} + AB \\ &= \bar{A}BCD + A(\bar{B} + B) \\ &= \bar{A}BCD + A \end{aligned}$$

Il est possible de minimiser encore plus cette expression en invoquant le théorème (15) qui établit que $x + \bar{x}y = x + y$. Dans notre cas précis, $x = A$ et $y = BCD$. Donc

$$z = \bar{A}BCD + A = BCD + A$$

Le circuit correspondant à cette expression finale est celui de la figure 4-8 c).

Cet exemple démontre parfaitement que la méthode de simplification peut devenir très laborieuse quand l'expression originale compte un grand nombre de termes. Ce n'est toutefois pas le cas avec la méthode des diagrammes de Karnaugh, comme vous le constaterez plus loin.

La construction du circuit définitif Dans les exemples de conception que nous venons de voir, le circuit définitif était construit au moyen de portes ET et OU. En fait, toutes les sommes de produits donnent lieu à un circuit comprenant une ou plusieurs portes ET pilotant une seule porte OU. L'une des raisons qui nous incitent à rechercher une somme de produits est la possibilité de la matérialiser en utilisant seulement des portes NON-ET et d'avoir un circuit, la plupart du temps, pas plus complexe que celui construit avec des portes ET et OU. Comme les portes NON-ET sont les portes de la famille logique TTL les plus répandues, il s'agit là d'une caractéristique non négligeable.

À titre d'illustration, voyons la figure 4-9 qui montre les circuits NON-ET équivalents aux circuits des figures 4-7 b) et 4-8 c). Procédez vous-mêmes à ces conversions en guise de révision de la marche à suivre exposée au chapitre 3.

Si on compare le circuit NON-ET avec l'autre circuit de la figure 4-9 a), on voit qu'ils ont la même structure, c.-à-d. que *chaque* porte du circuit original a été remplacée par une seule porte NON-ET. Cette correspondance n'est vérifiée que si le circuit original se présente sous la forme d'une somme de produits. La seule exception s'observe lorsque la somme de produits comprend un terme ayant une seule variable comme $z = A + BCD$ [circuit de la figure 4-9 b)]. Dans ce dernier cas, le circuit NON-ET exige une porte NON-ET supplémentaire qui agit comme INVERSEUR de l'entrée A.

Nous pouvons systématiser ainsi le processus de conversion d'un circuit *somme de produits* construit avec des portes ET et OU en un circuit construit avec des portes NON-ET.

1. Remplacez chaque porte ET, OU et INVERSEUR par une *seule* porte NON-ET.
2. Utilisez une porte NON-ET pour inverser toute variable unique qui alimente la porte OU.

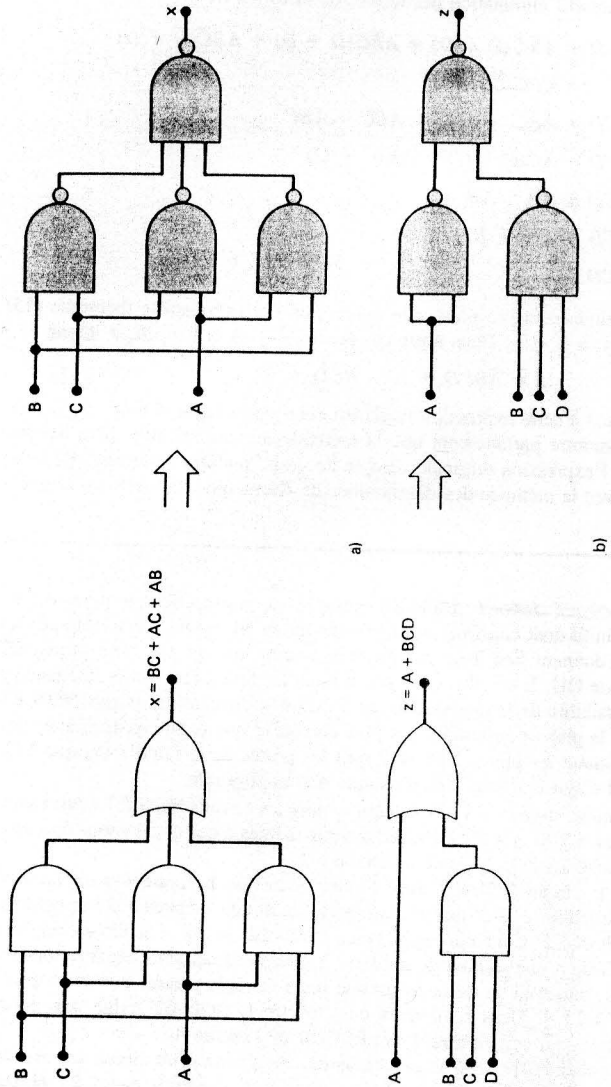


FIGURE 4-9 a) Le circuit de la figure 4-7 b) et son circuit équivalent construit avec des portes NON-ET; b) le circuit de la figure 4-8 c) et son circuit équivalent construit avec des portes NON-ET.

EXEMPLE 4-9A

Concevez un circuit logique dont les entrées P, Q, R sont telles que la sortie S est HAUTE quand $P = 0$ ou quand $Q = R = 1$.

Solution La table de vérité de cet énoncé est illustrée ci-après. Il y a cinq conditions d'entrée qui donnent une valeur HAUTE en sortie.

P	Q	R	S	
0	0	0	1	$\overline{P}\,\overline{Q}\,\overline{R}$
0	0	1	1	$\overline{P}\,\overline{Q}R$
0	1	0	1	$\overline{P}Q\overline{R}$
0	1	1	1	$\overline{P}QR$
1	0	0	0	
1	0	1	0	
1	1	0	0	
1	1	1	1	PQR

La somme de produits est donc:

$$S = \overline{P}\,\overline{Q}\,\overline{R} + \overline{P}\,\overline{Q}R + \overline{P}Q\overline{R} + \overline{P}QR + PQR$$

Commençons le processus de simplification en mettant en facteur $\overline{P}\,\overline{Q}$ des termes 1 et 2 et en mettant en facteur $\overline{P}Q$ des termes 3 et 4:

$$S = \overline{P}\,\overline{Q}(\overline{R} + R) + \overline{P}Q(\overline{R} + R) + PQR$$

À présent, éliminons les termes $\overline{R} + R$ puisqu'ils valent 1:

$$S = \overline{P}\,\overline{Q} + \overline{P}Q + PQR$$

Mettons en facteur $\overline{P}$ dans les termes 1 et 2, ce qui nous autorise à éliminer Q de ceux-ci:

$$S = \overline{P} + PQR$$

Appliquons dans ce cas le théorème (15) $x + \overline{x}y = x + y$, ce qui donne:

$$S = \overline{P} + QR$$

La matérialisation de cette expression avec des portes ET/OU est montrée à la figure 4-10 a).

EXEMPLE 4-9B

Réalisez avec des portes NON-ET le circuit de l'exemple 4-9A.

Solution Remplaçons chaque porte OU et ET de la figure 4-10 a) par une porte NON-ET et substituons à l'INVERSEUR un INVERSEUR porte NON-ET (marqué 1). En outre, comme l'entrée du haut de la porte OU est une monovariable ($\overline{P}$), il faut insérer un INVERSEUR porte NON-ET (marqué 2) devant cette entrée. Le circuit résultant est illustré à la figure 4-10 b). De toute évidence, on peut éliminer les INVERSEURS 1 et 2 (qui forment une double complémentation de P) pour obtenir le circuit définitif de la figure 4-10 c).

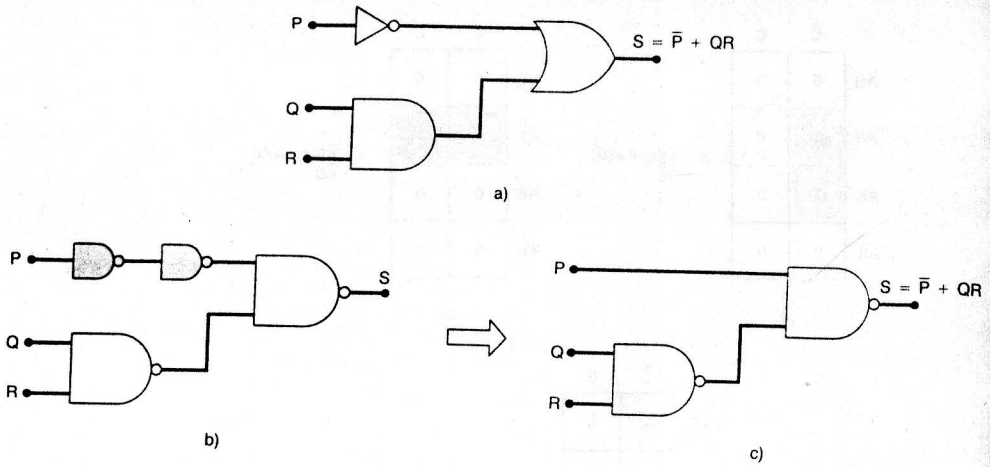


FIGURE 4-10 a) Matérialisation avec des portes ET/OU de la réponse de l'exemple 4-9A; b) Matérialisation avec des portes NON-ET du même circuit; c) simplification après élimination des deux INVERSEURS.

QUESTIONS DE RÉVISION

1. Écrivez la somme de produits correspondant à un circuit ayant quatre entrées et une sortie qui prend le niveau HAUT seulement quand l'entrée *A* en même temps que deux autres entrées sont au niveau BAS.
2. Construisez le circuit de l'expression de la question 1 en n'utilisant que des portes NON-ET. Indiquez le nombre de portes nécessaire.

4-5 LA MÉTHODE DES DIAGRAMMES DE KARNAUGH*

Le diagramme de Karnaugh est un outil graphique qui permet de simplifier de manière méthodique une équation logique ou le processus de passage d'une table de vérité à son circuit correspondant. Bien que les diagrammes de Karnaugh (à partir d'ici abrégé *diagrammes K*) soient applicables à des problèmes ayant un nombre quelconque de variables d'entrée, ils ne sont plus d'une grande utilité en pratique quand le nombre de variables dépasse six. En plus, dans cette section, nous n'allons pas aborder de problèmes ayant plus de quatre entrées, puisque ceux ayant cinq et six entrées sont des problèmes d'envergure qu'il est préférable de traiter avec un programme informatique.

La forme du diagramme de Karnaugh Le *diagramme K*, tout comme la table de vérité, est un instrument qui met en évidence les rapports entre les entrées logiques et la sortie recherchée. La figure 4-11 nous fait voir trois exemples de diagrammes de Karnaugh pour

* On peut omettre cette section sans nuire à la compréhension du reste de ce livre.

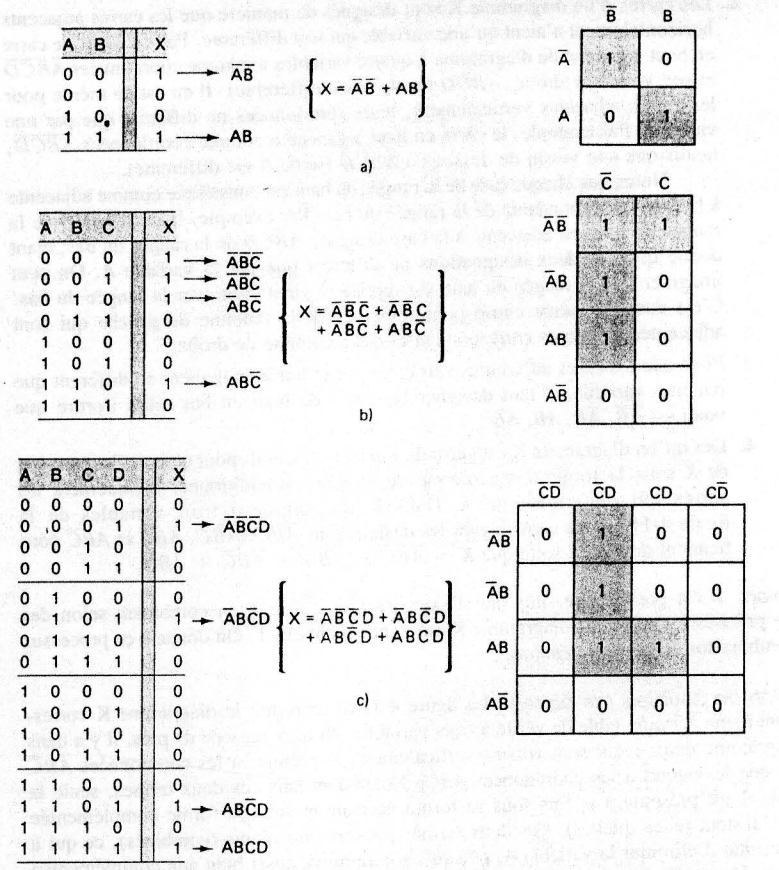


FIGURE 4-11 Les diagrammes de Karnaugh et les tables de vérité pour a) deux, b) trois et c) quatre variables.

deux, trois et quatre variables, ainsi que les tables de vérité correspondantes. Ces exemples mettent en lumière les points importants que voici:

1. La table de vérité donne la valeur de la sortie *X* pour chacune des combinaisons des valeurs d'entrée; par contre, le diagramme K organise l'information de manière différente. Chaque ligne de la table de vérité correspond à un carré du diagramme de Karnaugh. Par exemple, à la figure 4-11 a), la ligne $A = 0$ et $B = 0$ de la table de vérité est le carré $\overline{A}\,\overline{B}$ du diagramme K. Étant donné que pour cette ligne *X* vaut 1, on inscrit 1 dans ce carré. De même, on associe à la ligne $A = 1$ et $B = 1$ le carré qui a les coordonnées *AB*. Comme *X* vaut aussi 1 dans ce cas, on retrouve un 1 dans ce carré. Les autres carrés de ce diagramme K contiennent des 0. Le même raisonnement s'applique pour les diagrammes à trois et quatre variables.

2. Les carrés d'un diagramme K sont désignés de manière que les carrés adjacents horizontalement n'aient qu'une variable qui soit différente. Par exemple, le carré en haut à gauche du diagramme à quatre variables a comme coordonnées $\overline{A}\overline{B}\overline{C}\overline{D}$ et son voisin de droite, $\overline{A}\overline{B}\overline{C}D$ (seule D est différente). Il en est de même pour les carrés adjacents verticalement, leurs coordonnées ne diffèrent que par une variable. Par exemple, le carré en haut à gauche a comme coordonnées $\overline{A}\overline{B}\overline{C}\overline{D}$, tandis que son voisin du dessous a $\overline{A}B\overline{C}\overline{D}$ (seule B est différente).

   Notez que chaque case de la rangée du haut est considérée comme adjacente à la case correspondante de la rangée du bas. Par exemple, la case $\overline{A}\overline{B}CD$ de la rangée du haut est adjacente à la case désignée $A\overline{B}CD$ de la rangée du bas, étant donné que ces deux désignations ne diffèrent que par la variable A. On peut imaginer que la rangée du haut est repliée et vient continuer la rangée du bas. C'est aussi la même chose pour les cases de la colonne de gauche qui sont adjacentes aux cases correspondantes de la colonne de droite.

3. Pour que les cases adjacentes verticalement et horizontalement ne diffèrent que par une variable, il faut désigner les cases de haut en bas selon l'ordre que voici — $\overline{A}\overline{B}$, $\overline{A}B$, AB, $A\overline{B}$.

4. Dès qu'un diagramme K est garni de 1 et de 0, il s'agit, pour obtenir l'expression de X sous la forme d'une somme de produits, d'additionner logiquement les carrés qui renferment un 1. Dans le diagramme à trois variables de la figure 4-11 b), les carrés ayant les désignations $\overline{A}\overline{B}\overline{C}$, $\overline{A}B\overline{C}$, $\overline{A}B\overline{C}$ et $AB\overline{C}$ contiennent des 1, de sorte que $X = \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + \overline{A}B\overline{C} + AB\overline{C}$.

Réunion Il est possible de simplifier l'expression de sortie X en combinant selon des règles précises les carrés du diagramme K qui contiennent des 1. On donne à ce processus de combinaison le nom de *réunion*.

Réunion de doublets (de paires) La figure 4-12 a) reproduit le diagramme K correspondant à une certaine table de vérité à trois variables. Si on y regarde de près, il y a dans ce diagramme deux 1 qui sont voisins verticalement: le premier a les coordonnées $\overline{A}B\overline{C}$ tandis que le second a les coordonnées $AB\overline{C}$. Notez que dans ces deux termes, seule la variable A est présente à la fois sous sa forme normale et sous sa forme complémentée (B et C restent telles quelles). Ces deux termes peuvent être réunis (combinés), ce qui a pour résultat d'éliminer la variable A, puisqu'on la retrouve aussi bien non complémentée que complémentée. La preuve de cette dernière affirmation est simple, la voici:

$$\begin{aligned} X &= \overline{A}B\overline{C} + AB\overline{C} \\ &= B\overline{C}(\overline{A} + A) \\ &= B\overline{C}\,(1) = B\overline{C} \end{aligned}$$

Le même principe joue toujours pour tout doublet de 1 voisins verticalement ou horizontalement. On voit à la figure 4-12 b) deux 1 voisins horizontalement. Il est possible de réunir ces deux 1 et d'éliminer la variable C, celle qui est à la fois non complémentée et complémentée. Le résultat est alors $X = \overline{A}B$.

Voyons un autre exemple: dans un diagramme K, la ligne du haut et la ligne du bas sont considérées comme adjacentes, de sorte que dans le diagramme de la figure 4-12 c) deux 1 peuvent être réunis pour donner $\overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} = \overline{B}\overline{C}$.

La figure 4-12 d) montre un diagramme K qui possède deux doublets pouvant être réunis. D'abord il y a deux 1 dans la ligne du haut qui sont voisins horizontalement; ensuite, il y a deux 1 dans la ligne du bas qui sont également voisins, puisque dans un diagramme K la colonne de droite et la colonne de gauche sont considérées comme adjacentes. La

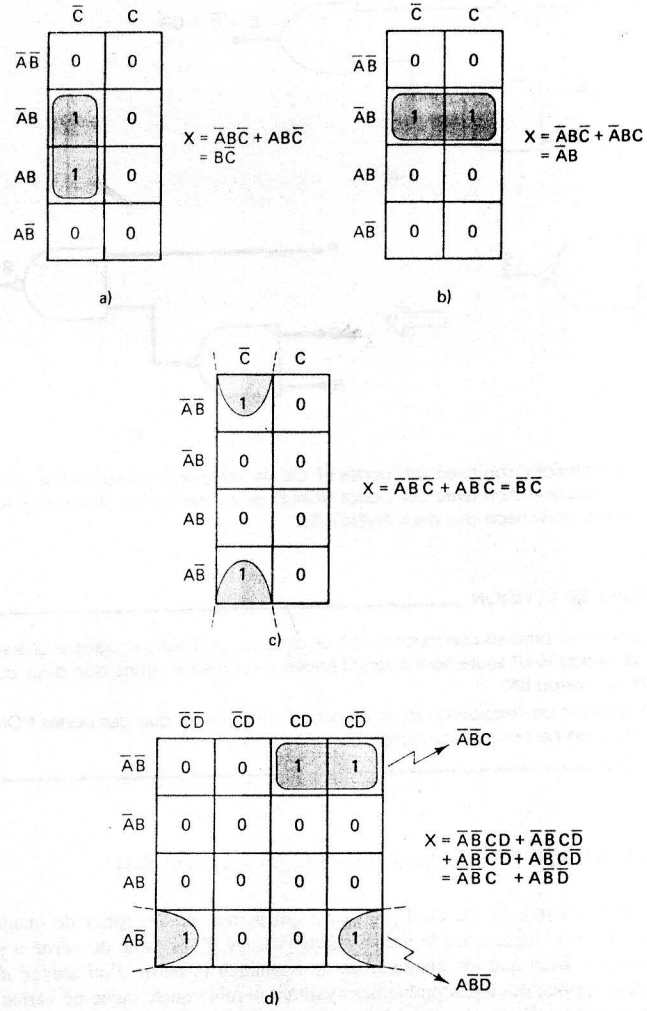


FIGURE 4-12 Exemples de réunion de doublets de 1 adjacents.

réunion des deux 1 de la ligne du haut élimine D (puisqu'il y a D et $\overline{D}$), pour donner $\overline{A}B\overline{C}$, tandis que la réunion des deux 1 de la ligne du bas élimine C pour donner $AB\overline{D}$. La valeur X finale est la somme logique de ces deux termes.

En résumé:

La réunion d'un doublet de 1 adjacents dans un diagramme K élimine la variable qui est à la fois non complémentée et complémentée.

Réunion de quartets (groupes de quatre) Il peut arriver qu'un diagramme K contienne quatre 1 qui soient adjacents, c.-à-d. qu'il contienne un *quartet* de 1 adjacents. La

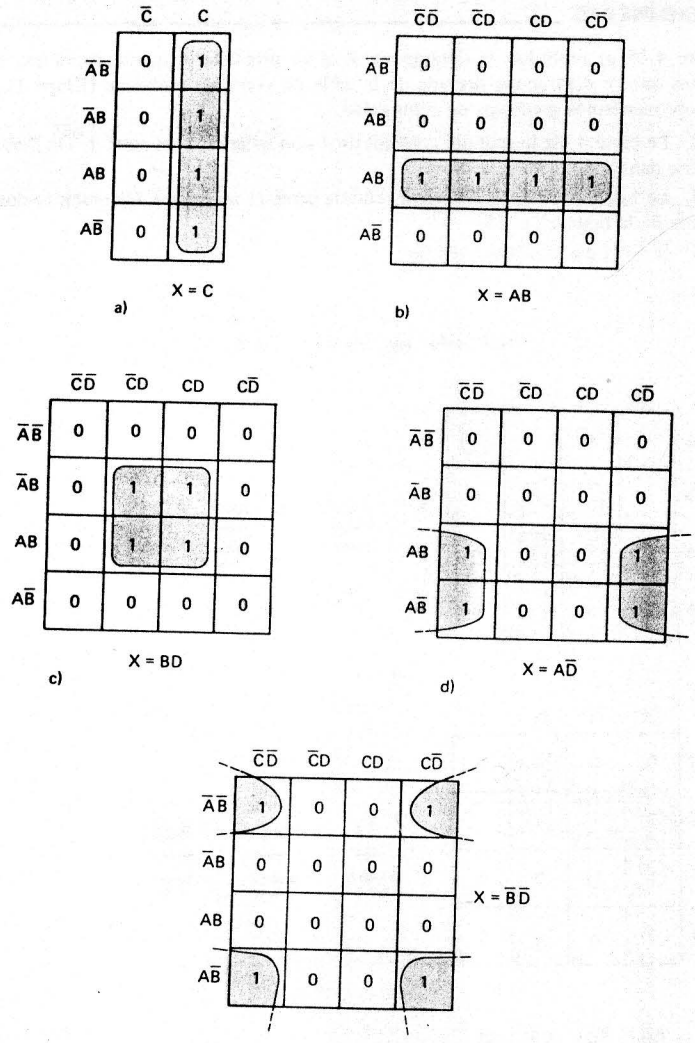


FIGURE 4-13 Exemple de réunion de quartets.

figure 4-13 regroupe plusieurs exemples de tels quartets. En a) les quatre 1 sont voisins verticalement, alors qu'en b), ils le sont horizontalement. Le diagramme de c) renferme quatre 1 arrangés en carré qui sont considérés adjacents les uns aux autres. Les quatres 1 de d), tout comme ceux de e), sont aussi adjacents, parce que, comme nous l'avons déjà dit, dans un diagramme de Karnaugh les lignes du haut et du bas et les colonnes droite et gauche sont considérées comme adjacentes.

Quand on réunit un quartet, on ne retrouve dans le terme résultant que les variables qui restent sous la même forme dans tous les carrés du quartet. Par exemple, en a), les coordonnées des quatre carrés sont $\overline{A}\overline{B}C$, $\overline{A}BC$, ABC et $A\overline{B}C$. L'examen de ces quatre termes nous apprend que seule la variable C demeure inchangée. En effet A et B apparaissent tous les deux sous la forme non complémentée et la forme complémentée. D'où le fait que l'expression définitive pour X est simplement $X = C$. En voici la démonstration:

$$\begin{aligned} X &= \overline{A}\overline{B}C + \overline{A}BC + ABC + A\overline{B}C \\ &= \overline{A}C(\overline{B} + B) + AC(B + \overline{B}) \\ &= \overline{A}C + AC \\ &= C(\overline{A} + A) = C \end{aligned}$$

Comme autre exemple, voyons la figure 4-13 d) où quatre carrés renferment des 1, soit $AB\overline{C}\overline{D}$, $A\overline{B}\overline{C}\overline{D}$, $AB\overline{C}D$ et $A\overline{B}CD$. L'examen de ces termes nous révèle que seules les variables A et $\overline{D}$ restent telles quelles, de sorte que l'expression simplifiée pour X est

$$X = A\overline{D}$$

La preuve de ceci s'obtient comme dans les calculs précédents.

Le lecteur est prié de vérifier que dans les autres cas de la figure 4-13, l'expression définitive est bien celle indiquée. Pour résumer:

La réunion d'un quartet de 1 adjacents élimine les deux variables qui sont à la fois non complémentées et complémentées.

Réunion d'octets (groupes de huit) Quand on réunit huit 1 adjacents, on dit qu'on réunit un octet de 1 adjacents. On peut voir à la figure 4-14 plusieurs exemples de réunion d'octets. La réunion d'un octet dans un diagramme K à quatre variables donne lieu à l'élimination de trois variables, car une seule variable demeure inchangée. Par exemple, observez bien les huit carrés réunis en a). Seule la variable B est toujours sous la même forme dans les huit termes, alors que toutes les autres sont non complémentées et complémentées. Aussi, pour ce diagramme $X = B$. Le lecteur peut vérifier les résultats des autres exemples de la figure 4-14.

En résumé:

La réunion d'un octet de 1 adjacents élimine les trois variables qui sont à la fois non complémentées et complémentées.

Le processus de simplification au complet Nous venons de voir comment la réunion de doublets, de quartets et d'octets de 1 adjacents dans un diagramme K aboutit à une expression simplifiée. Voici la règle générale valable pour toute réunion quel que soit le nombre de 1 adjacents: *Quand une variable est présente à la fois sous la forme non complémentée et la forme complémentée dans une réunion, cette variable est éliminée de l'expression. Seules apparaissent dans l'expression définitive les variables gardant la même forme dans tous les carrés d'une réunion.*

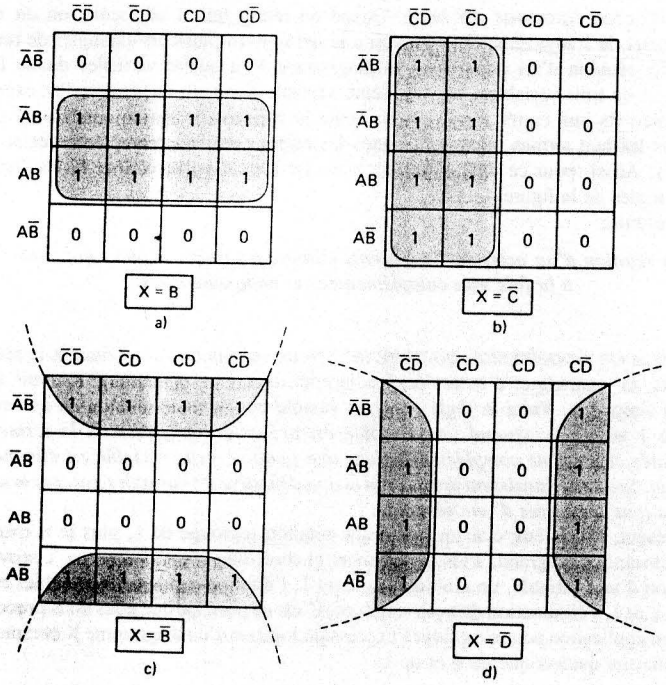
À présent, il doit être clair que plus une réunion regroupe de 1, plus le nombre de variables éliminées est grand. Plus précisément encore, une réunion de deux 1 provoque l'élimination d'une variable, un réunion de quatre 1, l'élimination de deux variables et une réunion de huit 1, l'élimination de trois variables. C'est ce principe que nous nous proposons de mettre en application pour simplifier l'expression logique d'un diagramme K comprenant une combinaison quelconque de 0 et de 1.

En premier lieu, énumérons les étapes de la marche à suivre, ensuite nous en verrons l'application dans quelques exemples. Voici les étapes à suivre pour simplifier une expression booléenne en recourant à la méthode des diagrammes de Karnaugh:

1. Dessinez le diagramme K et placez des 1 dans les carrés correspondant aux lignes de la table de vérité dont la sortie est 1. Mettez des 0 dans les autres carrés.
2. Étudiez le diagramme et repérez les 1 adjacents. Encerclez les 1, dit *isolés*, qui ne sont voisins d'aucun autre 1.
3. Ensuite, trouvez les 1 qui sont adjacents seulement à un autre 1. Réunissez ces doublets de 1 adjacents.
4. Réunissez tous les octets de 1 adjacents même si des 1 se trouvent déjà dans des doublets.
5. Réunissez tous les quartets de 1 adjacents qui ont au moins un 1 qui n'a pas déjà été regroupé.
6. Réunissez tous les doublets nécessaires pour inclure n'importe quel 1 non encore regroupé, en prenant soin d'utiliser le moins de réunions possibles.
7. Effectuez l'addition logique de tous les termes résultant des réunions.

Nous allons suivre et indiquer scrupuleusement ces étapes dans la résolution des exemples qui suivent. Dans chaque cas, nous allons obtenir l'expression logique la plus simple qui ait la forme d'une somme de produits.

FIGURE 4-14 Exemples de réunion d'octets.



EXEMPLE 4-10

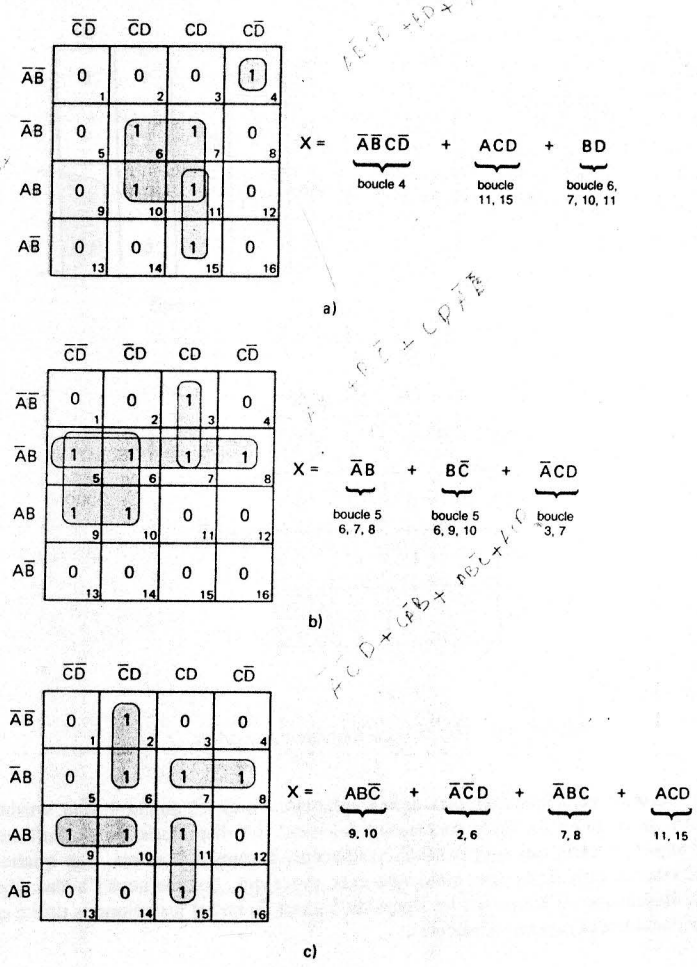
La figure 4-15 a) reproduit le diagramme K d'un problème à quatre variables. Nous supposons que ce diagramme découle de la table de vérité du problème (**Étape 1**). Les carrés sont numérotés pour plus de commodité.

Étape 2 Le carré 4 est le seul qui contient un 1 non adjacent à un autre 1. On l'entoure et on parle dans son cas de la boucle 4.

Étape 3 Le 1 du carré 15 est adjacent à celui du carré 11 *seulement*. On réunit ce doublet et on parle de la boucle 11, 15.

Étape 4 Il n'y a pas d'octets à réunir.

FIGURE 4-15 Exemples 4-10 à 4-12.



Étape 5 On a un quartet de 1 adjacents dans les carrés 6, 7, 10 et 11. Réunissons ce quartet (boucle 6, 7, 10 et 11). Notez que le 1 du carré 11 est repris même s'il se trouve déjà dans la boucle 11, 15.

Étape 6 Tous les 1 sont déjà réunis.

Étape 7 Chacune de ces boucles fournit un terme à l'expression définitive de X: la boucle 4 donne $\overline{A}\overline{B}C\overline{D}$, la boucle 11, 15, ACD (élimination de B) et la boucle 6, 7, 10 , 11, BD (après élimination de A et C).

EXEMPLE 4-11

Étudions le diagramme K de la figure 4-15 b). De nouveau, nous supposons que l'**Étape 1** est déjà effectuée.

Étape 2 Il n'y a pas de 1 isolé.

Étape 3 Le 1 du carré 3 est adjacent à celui du carré 1 *seulement*. Réunissons ce doublet (boucle 3, 7) qui donne, après élimination, le terme $\overline{A}CD$.

Étape 4 Il n'y a pas d'octet à réunir.

Étape 5 Il y a deux quartets de 1 adjacents: les carrés 5, 6, 7, 8 en forment un premier dont la réunion donne, après élimination, le terme $\overline{A}B$. Les carrés 5, 6, 9 et 10 forment le second. On est justifié de réunir ce quartet de 1 adjacents parce qu'il y en a deux qui ne sont pas encore regroupés. La réunion de ce quartet de 1 adjacents donne, aprés élimination, le terme $B\overline{C}$.

Étape 6 Tous les 1 sont déjà réunis.

Étape 7 Les termes produits par ces trois réunions sont additionnés logiquement pour constituer l'expression de X.

EXEMPLE 4-12

Analysons le diagramme K de la figure 4-15 c).

Étape 2 Il n'a pas de 1 isolé.

Étape 3 Le 1 du carré 2 est adjacent au 1 du carré 6 *seulement*. Ce doublet est réuni pour donner le terme $\overline{A}C\overline{D}$. De même, le 1 du carré 9 est adjacent au 1 du carré 10 seulement. La réunion de ce dernier doublet produit le terme $AB\overline{C}$. De la même façon, les boucles 7, 8 et 11, 15 donnent respectivement les termes $\overline{A}BC$ et ACD.

Étape 4 Il n'y a pas d'octets de 1 adjacents à réunir.

Étape 5 On remarque cependant un quartet de 1 adjacents; on ne le réunit pas parce que tous ses 1 sont déjà englobés.

Étape 6 Tous les 1 sont déjà englobés.

Étape 7 L'expression de X est celle indiquée sur la figure.

EXEMPLE 4-13

Étudions le diagramme K de la figure 4-16 a).

Étape 2 Il n'y a pas de 1 isolé.

Étape 3 Il n'y a pas de 1 adjacent à un seul autre 1.

Étape 4 Il n'y a pas d'octet.

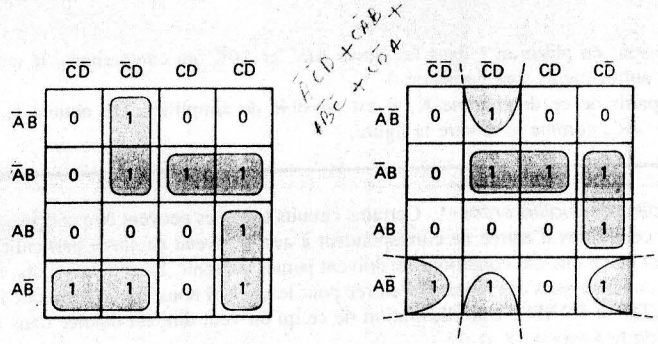


FIGURE 4-16 Le même diagramme K produit deux solutions aussi bonnes l'une que l'autre.

Étape 5 Il n'y a pas de quartet.

Étapes 6 et 7 Dans ce diagramme, il existe de nombreux doublets possibles. Quand nous réunissons des 1 adjacents, il faut utiliser le nombre minimal de boucles qui regroupent tous les 1. Dans le cas ci-dessus, il y a *deux* façons de faire les réunions qui produisent chacune quatre boucles de doublets. La figure 4-16 a) montre une solution et l'expression résultante, tandis que la figure 4-16 b) montre l'autre solution. Notez que ces deux expressions ont la même complexité, de sorte qu'aucune des deux n'est meilleure que l'autre.

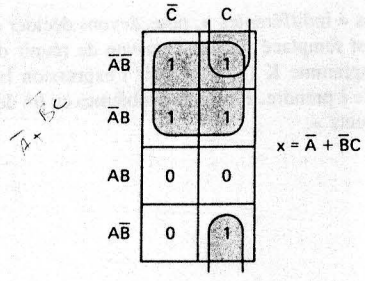
EXEMPLE 4-14

Au moyen d'un diagramme K, simplifiez l'expression $x = \overline{A}\overline{B}\overline{C} + \overline{B}C + \overline{A}B$.

Solution Voilà un problème pour lequel nous n'avons pas la table de vérité permettant de garnir le diagramme K. Nous devons remplir ce diagramme en considérant chacun des termes de cette expression et en plaçant un 1 dans les carrés dont les coordonnées incluent ce terme.

Le premier terme $\overline{A}\overline{B}\overline{C}$ nous informe qu'il y a un 1 dans le carré $\overline{A}\overline{B}\overline{C}$ (voir figure 4-17). Le second terme, $\overline{B}C$, indique que l'on place un 1 dans tous les carrés dont les coordonnées contiennent $\overline{B}C$, soit, sur la figure 4-17, les carrés $A\overline{B}C$ et $\overline{A}\overline{B}C$. Pour la

FIGURE 4-17 Exemple 4-14.



même raison, on place un 1 dans les carrés $\overline{A}BC$ et $\overline{A}B\overline{C}$ qui comprennent le terme $\overline{A}B$. Tous les autres carrés sont garnis de 0.

À partir de ce diagramme K, il est possible de simplifier. On obtient le résultat $x = \overline{A} + \overline{B}C$, comme le montre la figure.

*Les conditions « indifférentes »** Certains circuits logiques peuvent être conçus pour que certaines conditions d'entrée ne correspondent à aucun niveau de sortie particulier, principalement parce que ces conditions ne doivent jamais survenir. En d'autres mots, il existe certaines combinaisons des niveaux d'entrée pour lesquels il nous importe peu que la sortie soit HAUTE ou BASSE. Une illustration de ce qu'on veut dire est donnée dans la table de vérité de la figure 4-18 a).

Dans cette table aucune valeur de z ne figure pour les conditions $A, B, C = 1, 0, 0$ et $A, B, C = 0, 1, 1$. Au contraire, on a mis un x. Ce x signifie « condition indifférente ». Plusieurs raisons peuvent expliquer la présence de conditions « indifférentes », la plus courante étant que dans certaines situations ces combinaisons d'entrée ne peuvent jamais survenir; par conséquent, il est inutile de préciser pour elles une valeur de sortie.

Un concepteur de circuit est libre de mettre des 0 ou des 1 vis-à-vis des conditions « indifférentes » en vue de produire l'expression de sortie la plus simple. Par exemple, dans le diagramme K correspondant à la table de vérité de la figure 4-18 b) où x figure dans les carrés $A\overline{B}\overline{C}$ et $\overline{A}BC$, le concepteur peut décider de mettre un 1 dans le carré $A\overline{B}\overline{C}$ et un 0 dans le carré $\overline{A}BC$, puisque ceci produit un quartet de 1 adjacents qui peuvent être réunis pour produire $z = A$, comme on le voit à la figure 4-18 c).

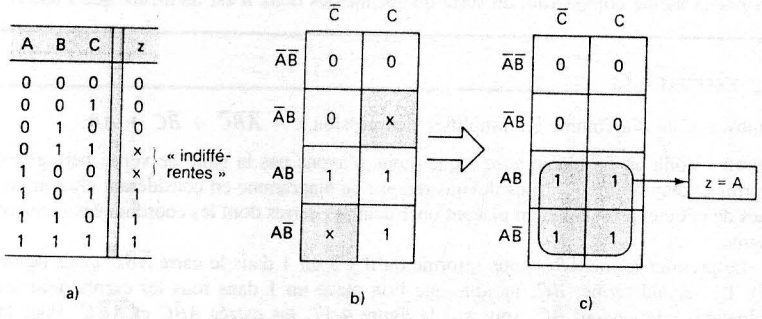


FIGURE 4-18 Il faut remplacer les x des conditions « indifférentes » par les valeurs (0 ou 1) qui permettent de constituer des réunions et d'obtenir l'expression la plus simple.

En présence de conditions « indifférentes », nous devons décider quel x de sortie est remplacé par un 0 et quel x est remplacé par un 1 en vue de réunir de la façon la plus efficace les 1 adjacents du diagramme K (pour obtenir l'expression la plus simple). La décision n'est pas toujours facile à prendre. Plusieurs problèmes en fin de chapitre abordent le traitement des cas « indifférents ».

* Dites aussi « indéterminées ».

RÉCAPITULATION

La méthode des diagrammes de Karnaugh présente plusieurs avantages comparativement à la méthode algébrique. C'est une méthode dont l'application se fait de manière méthodique en suivant des étapes bien définies, habituellement moins nombreuses surtout dans le cas d'expressions à plusieurs termes, et qui donne toujours l'expression minimale.

Malgré ces avantages, certains enseignants lui préfèrent la méthode algébrique qui exige une connaissance complète de l'algèbre booléenne et qui fait appel au raisonnement plutôt qu'à une démarche mécanique. Quoi qu'il en soit, chaque méthode a ses avantages et, même si la plupart des concepteurs de circuits logiques utilisent aussi bien l'une que l'autre, ce qui importe c'est d'en maîtriser au moins une puisque toutes les deux donnent des résultats valables.

QUESTIONS DE RÉVISION

1. Au moyen d'un diagramme de Karnaugh, simplifiez l'expression de l'exemple 4-7.
2. Au moyen d'un diagramme de Karnaugh, simplifiez l'expression de l'exemple 4-8. Voyez comment les diagrammes de Karnaugh facilitent le travail quand l'expression possède beaucoup de termes.
3. Dites ce qu'on entend par « conditions indifférentes ».

4-6 CIRCUITS OU EXCLUSIF ET NI EXCLUSIF

Deux circuits logiques spéciaux qui interviennent souvent dans les systèmes numériques: le circuit *OU exclusif* et le circuit *NI exclusif*.

OU exclusif Examinons le circuit logique de la figure 4-19 a). L'expression de sortie de ce circuit est:

$$x = \overline{A}B + A\overline{B}$$

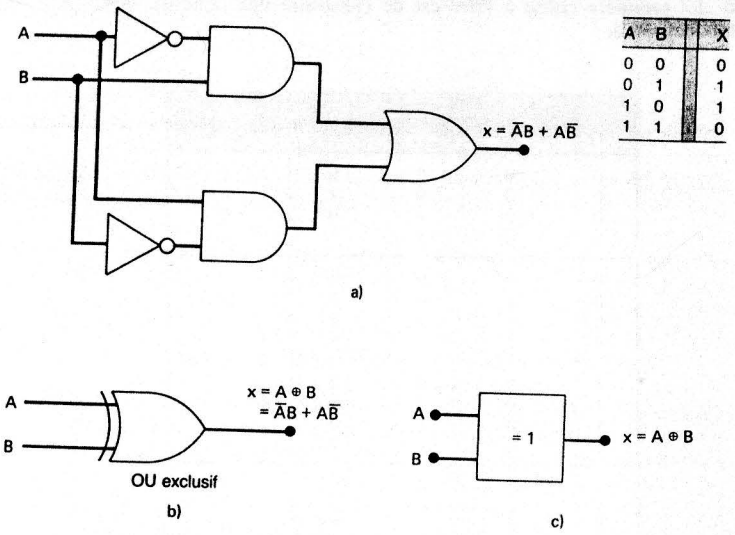
La table de vérité associée montre que $x = 1$ dans deux cas: $A = 0$, $B = 1$ (le terme $\overline{A}B$) et $A = 1$, $B = 0$ (le terme $A\overline{B}$). Cela signifie que ***ce circuit produit en sortie un niveau HAUT quand les signaux sur les deux entrées sont opposés***. Il s'agit ici du circuit OU exclusif.

Cet agencement de portes logiques est très courant et d'une grande utilité dans certaines applications. En fait, le circuit OU exclusif possède son propre symbole, celui illustré à la figure 4-19 b). Quand vous rencontrez ce symbole, vous devez considérer qu'il représente toutes les portes constituant un circuit OU exclusif, donc qu'il a la même expression et la même table de vérité. On donne souvent à ce circuit le nom de *porte* OU exclusif. Le nouveau symbole IEEE/ANSI d'une porte OU exclusif est représenté à la figure 4-19 c). Il signale que la sortie est vraie au niveau HAUT *si* une entrée *seulement* est au niveau HAUT.

Une porte OU exclusif n'a toujours que deux entrées. On veut dire par là qu'il n'existe pas de portes OU exclusif à trois ou quatre entrées. Ces deux entrées sont combinées pour que $x = \overline{A}B + A\overline{B}$. Parfois, on abrège cette expression ainsi:

$$x = A \oplus B$$

où le signe $\oplus$ représente l'opération OU exclusif.



A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

FIGURE 4-19 a) Le circuit OU exclusif et sa table de vérité; b) le symbole traditionnel de la porte OU exclusif; c) son symbole IEEE/ANSI.

Voici un résumé des caractéristiques de la porte OU exclusif:

1. N'a que deux entrées et sa sortie a pour valeur:

$$x = \overline{A}B + A\overline{B} = A \oplus B$$

2. Sa sortie est au niveau HAUT seulement quand les deux entrées sont à des niveaux *différents*.

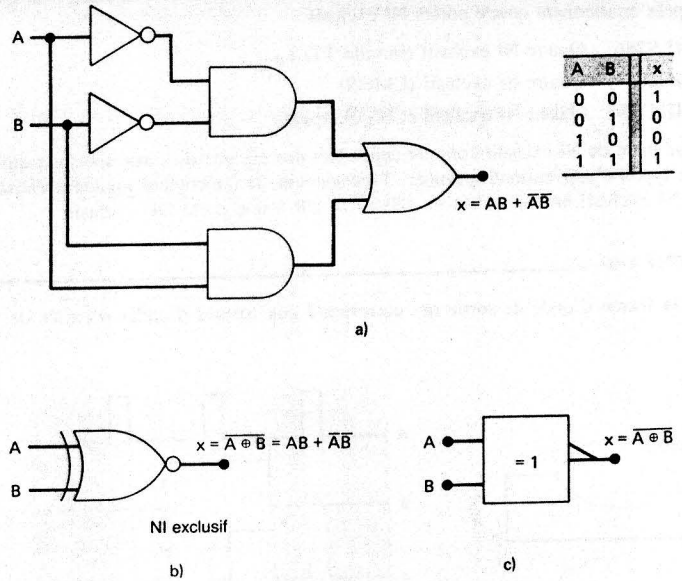
On trouve sur le marché plusieurs circuits intégrés qui contiennent des portes OU exclusif. Les boîtiers énumérés ci-après renferment chacun quatre portes OU exclusif.

- **7486** Quatre OU exclusif (famille TTL).
- **74C86** Quatre OU exclusif (famille CMOS).
- **74HC86** Quatre OU exclusif (CMOS rapides).

Ni exclusif Le circuit NI exclusif a un fonctionnement exactement opposé à celui du circuit OU exclusif. On peut voir à la figure 4-20 a) un circuit NI exclusif et sa table de vérité associée. L'expression de sortie de ce dernier circuit est:

$$x = AB + \overline{A}\overline{B}$$

ce qui indique, tout comme la table de vérité, que x prend la valeur 1 dans deux cas: $A = B = 1$ (le terme AB) et $A = B = 0$ (le terme $\overline{A}\overline{B}$). En clair, ***ceci signifie qu'un tel circuit produit un niveau de sortie HAUT quand les deux entrées sont identiques.***



A	B	x
0	0	1
0	1	0
1	0	0
1	1	1

FIGURE 4-20 a) Le circuit NI exclusif et sa table de vérité; b) le symbole traditionnel de la porte NI exclusif; son symbole IEEE/ANSI

Dès lors, il est évident que la sortie du circuit NI exclusif est exactement l'opposé de celle du circuit OU exclusif. Le symbole de la porte NI exclusif est construit à partir du symbole de la porte OU exclusif auquel est ajouté un petit rond en sortie [figure 4-20 b)]. Le symbole IEEE/ANSI contient le petit triangle devant la sortie de NI exclusif. Les deux symboles font savoir que le niveau vrai BAS ne se réalise que quand une entrée *seulement* est HAUTE.

La porte NI exclusif ne compte que *deux* entrées qui sont combinées pour obtenir en sortie:

$$x = AB + \overline{A}\overline{B}$$

La façon abrégée d'écrire l'expression de sortie du NI exclusif est:

$$x = \overline{A \oplus B}$$

ce qui est l'inverse de l'opération OU exclusif. Voici un résumé des caractéristiques de la porte NI exclusif:

1. N'a que deux entrées et sa sortie a pour valeur:

$$x = AB + \overline{A}\overline{B} = \overline{A \oplus B}$$

2. Sa sortie est au niveau HAUT seulement quand les deux entrées sont à des niveaux *identiques*.

Plusieurs CI sur le marché contiennent des portes NI exclusif. Ceux que l'on mentionne ci-après contiennent quatre portes NI exclusif.

- **74LS266** Quatre NI exclusif (famille TTL)
- **74C266** Quatre Ni exclusif (CMOS)
- **74HC266** Quatre NI exclusif (CMOS rapide)

Chaque puce de NI exclusif contient cependant des circuits de sortie spéciaux qui la limite à des types d'applications spéciaux. Très souvent, le concepteur logicien réalisera la fonction NI exclusif en connectant un INVERSEUR à une porte OU exclusif.

EXEMPLE 4-15

Déterminez la forme d'onde de sortie qui correspond aux formes d'ondes d'entrée de la figure 4-21.

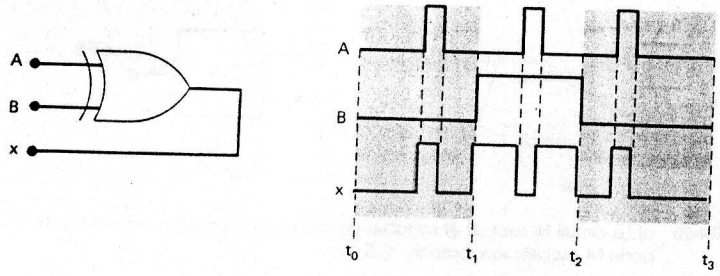


FIGURE 4-21 Exemple 4-15

Solution On trouve la forme d'onde de sortie en exploitant le fait que la sortie d'une porte OU exclusif est HAUTE seulement quand les signaux à ses entrées sont différents. La forme d'onde de sortie qui en résulte affiche plusieurs points intéressants:

1. La forme d'onde de x est identique à celle du signal A pendant le temps où $B = 0$, soit les périodes $t_0 - t_1$ et $t_2 - t_3$.
2. La forme d'onde de x est l'*inverse* de celle du signal A pendant le temps où $B = 1$, soit la période $t_1 - t_2$.
3. Ces deux observations nous montrent qu'une porte OU exclusif peut être utilisée comme INVERSEUR *commandé*, c.-à-d. qu'on peut choisir une des entrées pour décider si on inverse ou non l'autre entrée. C'est une propriété utile dans certaines applications.

EXEMPLE 4-16

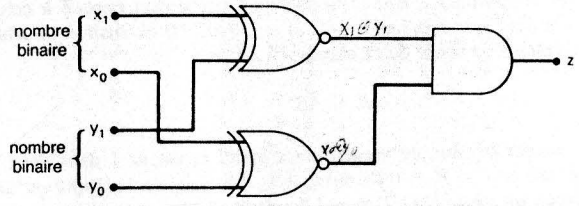
x_1x_0 représente un nombre binaire à 2 bits qui peut prendre l'une des quatre valeurs 00, 01, 10, 11; par exemple quand $x_1 = 1$ et que $x_0 = 0$, cela correspond au nombre 10... De la même manière y_1y_0 figure aussi un nombre binaire à 2 bits. Concevez un circuit logique tel que sa sortie prend la valeur 1 quand les deux nombres x_1x_0 et y_1y_0, les entrées de ce circuit, sont *égaux*.

Solution La première chose à faire est de construire une table de vérité pour les 16 conditions d'entrée.

x_1	x_0	y_1	y_0	z (sortie)
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

La valeur z est à 1 quand les valeurs x_1x_0 et y_1y_0 concordent, c.-à-d. quand $x_1 = y_1$ et $x_0 = y_0$. Dans la table on voit qu'il y a quatre combinaisons de ce genre. On pourrait pousuivre la résolution de ce problème en suivant la procédure habituelle, soit obtenir la somme de produits pour z, la simplifier, le cas échéant, puis construire le circuit résultant. Toutefois, la nature même de ce problème impose tout naturellement une solution avec des portes NI exclusif. En fait, un peu de raisonnement devrait nous permettre de trouver rapidement une solution simple. Reportez-vous à la figure 4-22; dans ce schéma x_1 et y_1 alimentent une porte NI exclusif et x_0 et y_0 alimentent une autre porte NI exclusif. La sortie de chacune de ces portes prend le niveau HAUT seulement quand ses entrées sont égales. Donc quand $x_1 = y_1$ et $x_0 = y_0$ les sorties des deux portes NI exclusif sont à 1. C'est la condition que nous cherchions à réaliser, puisqu'elle est équivalente à dire que les deux nombres sont égaux. Il suffit alors de multiplier logiquement (porte ET) ces deux dernières sorties pour obtenir le circuit recherché.

FIGURE 4-22 Circuit détecteur de l'égalité de deux nombres binaires.



EXEMPLE 4-17

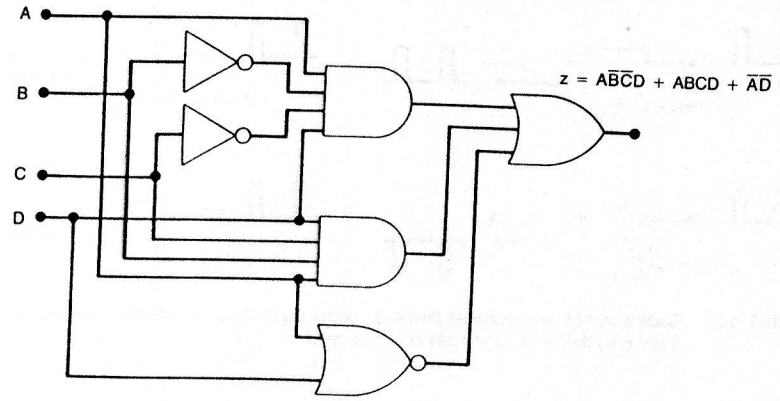
Au moment de la simplification de l'expression de la sortie d'un circuit logique combinatoire, il pourra arriver qu'après mise en facteur vous puissiez identifier une fonction OU exclusif ou NI exclusif. Cela peut vous conduire à recourir à des portes OU exclusif ou NI exclusif dans la matérialisation du circuit final. À titre d'illustration, simplifiez le circuit de la figure 4-23 a).

Solution L'expression non simplifiée de la sortie de ce circuit est

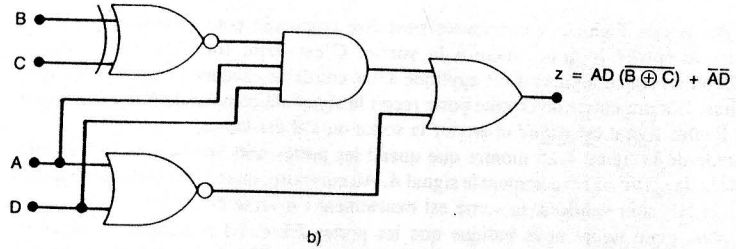
$$Z = ABCD + A\overline{B}\overline{C}D + \overline{A}\overline{D}$$

Il est possible de mettre en facteur AD dans les deux premiers termes:

$$Z = AD(BC + \overline{B}\overline{C}) + \overline{A}\overline{D}$$



a)



b)

FIGURE 4-23 Exemple 4-17; voyez une porte NI exclusif qui figure dans le circuit simplifié.

À prime abord, on sera porté à penser que l'expression entre parenthèses peut être remplacée par 1. Mais c'est faut, car c'est $BC + \overline{B}\overline{C}$ qui vaut 1. En fait, vous reconnaissez cette expression comme celle de l'opération NI exclusif portant sur B et C. Cette constatation peut être mise à profit pour refaire le circuit de la figure 4-23b. Ce nouveau circuit est beaucoup plus simple que l'original, puisqu'il utilise des portes ayant moins d'entrées et qu'il compte deux INVERSEURS en moins.

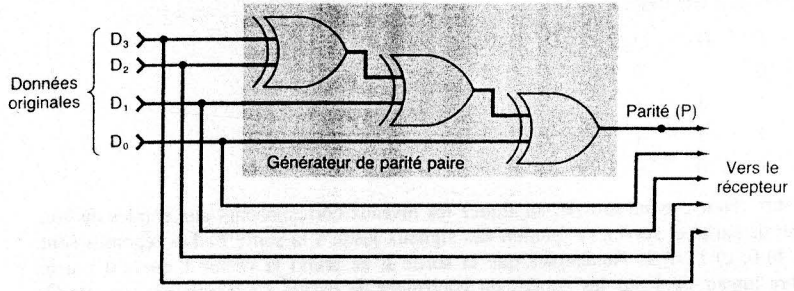
QUESTIONS DE RÉVISION

1. À l'aide de l'algèbre booléenne, prouvez que l'expression de sortie d'un NI exclusif est exactement l'opposé de l'expression de sortie d'un OU exclusif.
2. Déterminez la sortie d'une porte NI exclusif quand un signal logique et son complément sont appliqués à ses entrées?
3. Un concepteur de circuits logiques a besoin d'un inverseur mais ne possède qu'une porte OU exclusif dans une puce 7486. A-t-il besoin d'une autre puce?

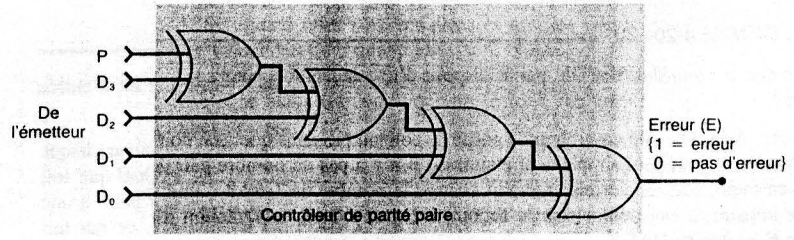
4-7 GÉNÉRATEUR ET CONTRÔLEUR DE PARITÉ

Au chapitre 2, nous avons vu qu'un émetteur peut joindre un bit de parité aux données qu'il transmet avant de les acheminer au récepteur. Nous avons également appris que le récepteur grâce à cette astuce pouvait détecter les erreurs monobits survenant durant la transmission. La figure 4-24 nous fait voir un exemple de circuit logique pouvant servir à la production et au contrôle de la parité. Cet exemple particulier porte sur des mots de quatre bits et concerne la parité paire. On peut adapter ces solutions telles quelles à la parité impaire et au nombre de bits que l'on veut...

FIGURE 4-24 Des portes OU exclusif sont utilisées pour réaliser concrètement un générateur et un contrôleur de parité paire.



a)



b)

À la figure 4-24 a), le groupe de données à transmettre est appliqué aux entrées d'un circuit générateur de parité, qui produit en sortie un bit de parité paire, P. Ce bit de parité est communiqué au récepteur en même temps que les données binaires, soit un total de cinq bits. À la figure 4-24 b), on peut voir ces cinq bits (donnée + parité) appliqués aux entrées du circuit de contrôle de la parité du récepteur, circuit qui fournit en sortie un signal d'erreur, E, qui annonce la présence ou l'absence d'une erreur monobit.

Cela ne devrait pas vous surprendre de retrouver des portes OU exclusif dans ces circuits, étant donné qu'une porte OU exclusif opère de telle manière qu'elle produit un 1 si un nombre impair de ses entrées sont à 1 et un 0 si un nombre pair de ses entrées sont à 1.

EXEMPLE 4-18

Déterminez la sortie du générateur de parité pour chacun des groupes binaires $D_3D_2D_1D_0$ suivants: a) 0111; b) 1001; c) 0000; d) 0100.

Solution Pour chaque groupe, appliquez les niveaux correspondants aux entrées du générateur de parité et suivez l'évolution des signaux dans chacune des portes jusqu'à la sortie P. Les résultats sont: a) 1; b) 0; c) 0; et d) 1. Remarquez que P ne vaut 1 que s'il y a dans la donnée originale un nombre impair de 1. Alors, le nombre total de 1 envoyés au récepteur (donnée + parité) est toujours pair.

EXEMPLE 4-19

Déterminez la sortie du contrôleur de parité pour chacun des groupes binaires suivants émanant de l'émetteur.

	P	D_3	D_2	D_1	D_0
a)	0	1	0	1	0
b)	1	1	1	1	0
c)	1	1	1	1	1
d)	1	0	0	0	0

Solution Pour chaque groupe, appliquez les niveaux correspondant aux entrées du contrôleur de parité et suivez l'évolution des signaux jusqu'à la sortie E. Les réponses sont: a) 0; b) 0; c) 1; d) 1. Remarquez que la sortie E ne prend la valeur 1 que s'il y a un nombre impair de 1 sur les entrées du contrôleur de parité. Ce 1 annonce une erreur, puisqu'il s'agit de surveiller la parité paire.

EXEMPLE 4-20

Est-ce que le contrôleur de parité dispose d'une indication pour connaître le bit d'entrée erroné?

Solution Non. Un contrôleur de parité ne connaît pas dans quel état doit se trouver chaque bit d'entrée; il peut simplement établir qu'il n'y a pas un nombre pair de 1. Quel que soit le bit en faute, une erreur monobit change le nombre de 1 total d'une valeur paire à une valeur impaire (à cause soit de l'ajout d'un 1 ou de la soustraction d'un 1), ce qui fait passer E au niveau HAUT.

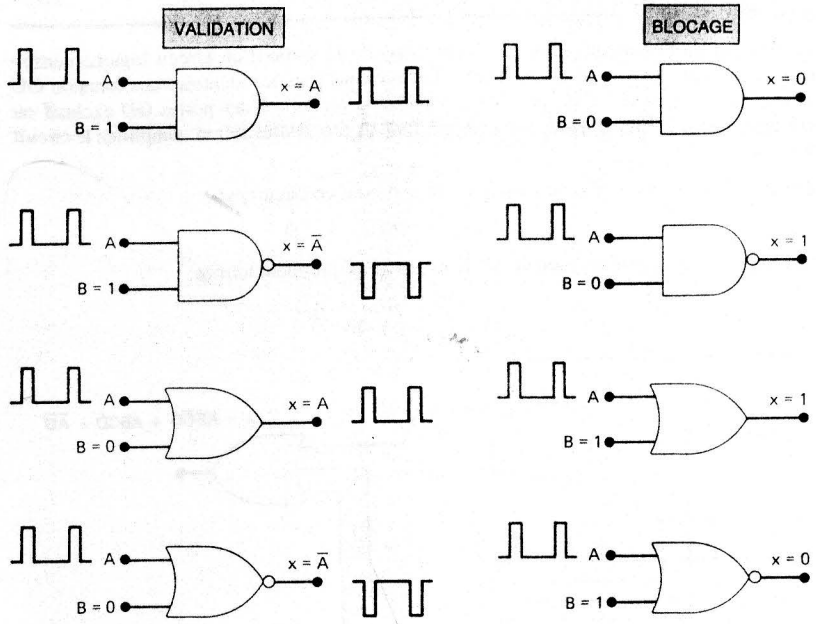


FIGURE 4-25 Quatre portes élémentaires peuvent valider ou bloquer le passage d'un signal A selon la valeur du signal de commande B.

4-8 CIRCUITS DE BLOCAGE

Chacune des portes logiques élémentaires peut être employée pour autoriser le passage d'un signal d'entrée logique jusqu'à la sortie. C'est cette fonction qu'illustre la figure 4-25 où un signal logique A est appliqué à une entrée de chacune des portes logiques élémentaires. L'autre entrée de chaque porte reçoit le signal de commande B dont le niveau stipule si l'autre signal est *validé* et atteint la sortie ou s'il est *bloqué*.

L'étude de la figure 4-25 montre que quand les portes non inverseuses (ET et OU) sont validées, la sortie suit exactement le signal A. Au contraire, quand les portes inverseuses (NON-ET et NI) sont validées, la sortie est exactement l'inverse du signal A.

De plus, cette figure nous indique que les portes ET et NI présentent toujours en sortie une valeur BASSE quand elles sont bloquées et que les portes OU et NON-ET présentent toujours une valeur HAUTE quand elles sont bloquées.

En conception de circuits numériques, il arrive souvent que le passage d'un signal logique soit validé ou bloqué selon les conditions présentes à des entrées de commande. Voici quelques exemples de cela.

EXEMPLE 4-21

Concevez un circuit logique qui autorise le passage d'un signal logique jusqu'à la sortie quand les entrées de commande B et C sont toutes les deux au niveau HAUT; dans les autres cas, la sortie est toujours au niveau BAS.

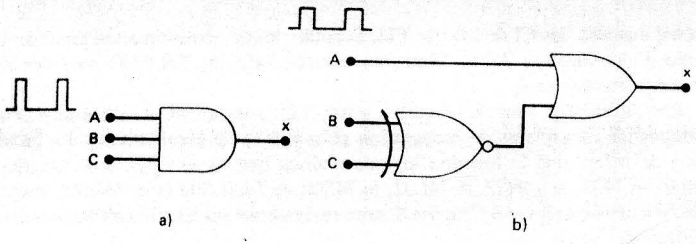


FIGURE 4-26 Exemples 4-21 et 4-22.

Solution On doit employer une porte ET parce que le signal ne doit pas être inversé et que la valeur de sortie durant le blocage est le niveau BAS. Comme la condition de validation est $B = C = 1$, il nous faut une porte ET à trois entrées, comme à la figure 4-26 a).

EXEMPLE 4-22

Concevez un circuit logique qui autorise le passage d'un signal jusqu'à la sortie seulement quand un des signaux de commande, mais pas les deux, est au niveau HAUT; dans les autres cas, la sortie reste au niveau HAUT.

Solution Le circuit recherché est illustré à la figure 4-26 b). On a utilisé une porte OU parce que la sortie durant le blocage doit rester à 1 et que le signal quand il passe ne doit pas être inversé. Les signaux de commande sont combinés dans une porte NI exclusif qui produit un signal BAS quand les deux entrées sont différentes. Ce signal BAS alimente la porte OU et la valide. Quand les deux signaux de commande sont identiques, la sortie de la porte NI exclusif est haute et la porte OU est bloquée.

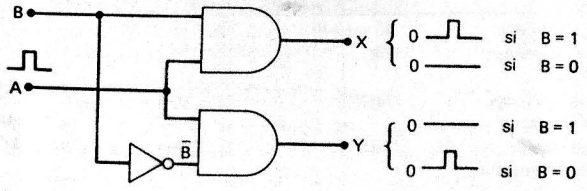
EXEMPLE 4-23

Concevez un circuit logique ayant une entrée A, une entrée de commande B et des sorties X et Y tel que:

1. Quand $B = 0$, la sortie Y suit le signal A et la sortie X est 0.
2. Quand $B = 1$, la sortie Y est 0 et la sortie X suit le signal A.

Solution Les deux sorties doivent rester à 0 dans leur condition respective de blocage et suivre le signal d'entrée dans leur condition respective de validation. Il faut donc une porte ET pour chaque sortie. Comme X est validé quand $B = 1$, sa porte ET est commandée par B (voir figure 4-27). Comme Y est validé quand $B = 0$, sa porte ET est commandée par $\overline{B}$.

FIGURE 4-27 Exemple 4-23.



Un tel circuit est appelé circuit d'*aiguillage d'impulsion* parce qu'il dirige l'impulsion vers une sortie ou une autre selon la valeur de B.

QUESTIONS DE RÉVISION

1. Concevez un circuit logique à trois entrées tel que sa sortie soit BASSE seulement quand A est HAUT et que B et C sont différents.
2. Nommez les portes qui ont toujours leur sortie à 1 quand elles sont bloquées.
3. Nommez les portes qui fournissent toujours l'inverse du signal quand elles sont validées.

4-9 CARACTÉRISTIQUES DE BASE DES CIRCUITS NUMÉRIQUES

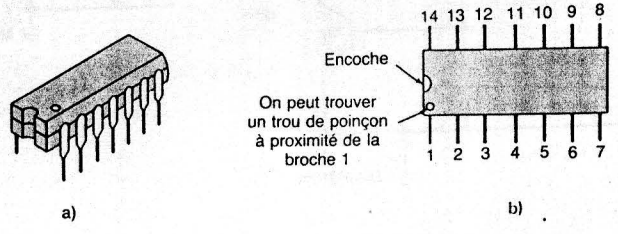
Les circuits intégrés numériques (CI) sont un regroupement de résistances, de diodes et de transistors fabriqués directement dans un matériau semiconducteur (habituellement du silicium), appelé *substrat* et que l'on désigne couramment du nom de « puce ». La puce est encapsulée dans un boîtier protecteur de plastique ou de céramique duquel émanent des broches servant à la connexion de la puce à d'autres dispositifs. Le boîtier le plus répandu est le *DIP* (deux rangées de broches), illustré à la figure 4-28. Les broches sont numérotées de gauche à droite quand on regarde le dessus du boîtier en partant d'une encoche ou d'un trou de poinçon repère. Le DIP montré est un boîtier à 14 broches. On trouve aussi des boîtiers à 16, 20, 24, 28, 40 et 64 broches.

Les CI numériques sont souvent désignés selon la complexité du circuit qu'ils portent, complexité mesurée par le nombre de portes logiques équivalentes réalisées dans le substrat. On retient habituellement cinq niveaux de complexité, dont voici les définitions:

COMPLEXITÉ	NOMBRE DE PORTES
Intégration à petite échelle (SSI)	Moins de 12
Intégration à moyenne échelle (MSI)	12 à 99
Intégration à grande échelle (LSI)	100 à 9999
Intégration à très grande échelle (VLSI)	10 000 à 99 999
Intégration à ultra grande échelle (ULSI)	Plus de 100 000

Tous les CI particuliers cités au Chapitre 3 et dans le présent chapitre sont des puces SSI, celles ayant le moins de portes. Dans les systèmes numériques modernes, les puces

FIGURE 4-28 a) Boîtier à deux rangées de broches; b) vue de dessus et numéros des broches.



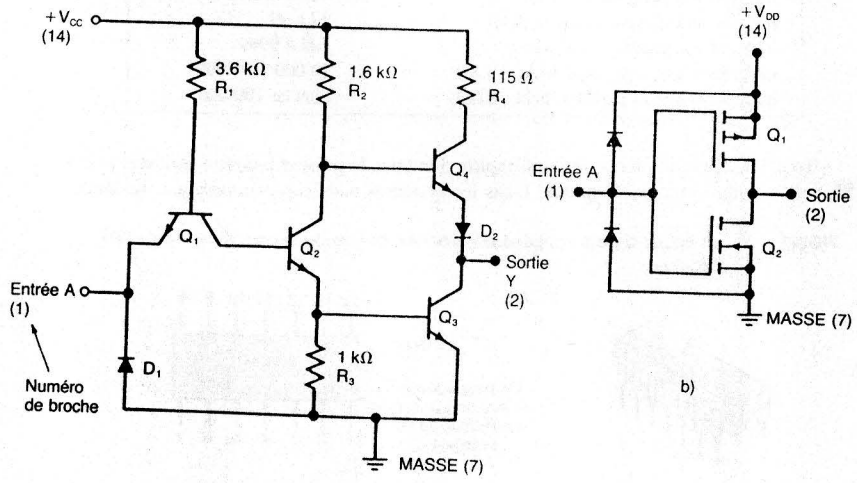
MSI (intégration à moyenne échelle) et les puces LSI (intégration à grande échelle) effectuent la plupart des fonctions qui auparavant nécessitaient plusieurs cartes de circuits remplies de composants SSI. Malgré cette substitution, les SSI restent encore utilisés comme « interfaces » entre ces puces complexes. Habituellement, de petites combinaisons de portes discrètes sont construites afin de connecter des CI plus grands à des dispositifs externes. C'est pourquoi il est indispensable de savoir comment analyser, concevoir, tester et dépanner des circuits combinatoires simples.

CI numériques bipolaires et unipolaires Les CI numériques peuvent aussi être catégorisés selon le type du principal composant électronique que l'on retrouve dans ses circuits. Les *CI bipolaires* sont ceux dont la construction est basée sur le transistor à jonction bipolaire (NPN et PNP). Les *CI unipolaires* sont ceux qui utilisent comme principal élément les transistors à effet de champ unipolaire (MOSFET à canal P et à canal N).

La famille de CI numériques bipolaires la plus répandue est la famille TTL (transistor-transistor logic). La figure 4-29 a) nous fait voir un INVERSEUR standard en technologie TTL. Remarquez la présence des nombreux transistors bipolaires. La famille TTL domine particulièrement les dispostifs SSI et MSI et occupe la tête depuis très longtemps. Sa prépondérance dans le domaine des circuits SSI et MSI est cependant de plus en plus contestée par la famille logique CMOS (MOS complémentaire), une catégorie du groupe plus général des CI numériques unipolaires. La figure 4-29 b) nous fait voir un INVERSEUR CMOS standard fabriqué avec des MOSFET à enrichissement. Plus loin, nous étudierons en détail les circuits TTL et CMOS.

Famille TTL La famille logique TTL telle qu'elle existe actuellement regroupe plusieurs sous-familles ou séries. Le tableau 4-1 énumère chacune des série TTL accompagnée du préfixe figurant dans le code du CI et indiquant que ce circuit fait partie de cette série. Par exemple, les CI qui font partie de la série TTL standard ont comme numéro d'identification 74. Quand on parle de 7402, 7438 et 74123, on sait qu'il s'agit de CI TTL standard. De

FIGURE 4-29 a) Circuit INVERSEUR TTL; b) circuit INVERSEUR CMOS. Les numéros des broches sont entre parenthèses.



la même manière, les CI de la série TTL Schottky faible consommation possèdent comme numéro d'identification 74LS. Donc, les circuits 74LS38, 74LS123 sont des exemples d'éléments de cette série.

Les différences entre les diverses séries TTL sont de nature électrique, comme la consommation, les retards de propagation et la vitesse de commutation. Le brochage du boîtier, de même que la fonction logique réalisée par les circuits, sont identiques. Par exemple, le 7402, le 74H02, le 74L02, le 74S02, le 74ALS02 et le 74AS02 sont tous des portes NI à deux entrées. Au Chapitre 8, nous reviendrons sur les caractéristiques électriques des différentes séries TTL.

TABLEAU 4-1 Diverses séries de la famille de circuits logiques TTL

Séries TTL	Préfixe	Exemple de CI
TTL standard	74	7404 (six INVERSEURS)
TTL rapide	74H	74H04 (six INVERSEURS)
TTL faible consommation	74L	74L04 (six INVERSEURS)
TTL Schottky	74S	74S04 (six INVERSEURS)
TTL Schottky faible consommation	74LS	74LS04 (six INVERSEURS)
TTL Schottky avancé	74AS	74AS04 (six INVERSEURS)
TTL Schottky avancé faible consommation	74ALS	74ALS04 (six INVERSEURS)

Famille CMOS Dans la famille CMOS, il y a aussi plusieurs séries. Celles-ci sont énumérées au tableau 4-2. Les séries 4000 et 14000 sont d'anciennes séries CMOS. Ces séries réalisent les mêmes fonctions logiques que celles de la famille TTL, mais elles n'ont pas été conçues pour avoir un brochage compatible avec les éléments TTL. Par exemple, la puce à quatre NI 4001 renferme quatre portes NI, comme la puce TTL 7402; cependant, les entrées et les sorties dans la puce CMOS n'ont pas les mêmes numéros de broche que les signaux correspondant de la puce TTL.

Les séries 74C, 74HC, 74HCT, 74AC et 74ACT sont des séries CMOS lancées il y a peu de temps, les deux dernières étant les plus récentes. Le brochage de toutes ces séries est compatible avec les éléments TTL numérotés de façon similaire. Par exemple, le 74C02, le 74HC02 et le 74HCT02 ont tous leurs broches compatibles avec le 7402, le 74H02, le 74LS02, etc. Les séries 74HC et 74HCT fonctionnent à une vitesse supérieure à celle des dispostifs 74C. Les séries 74HCT ont été conçues pour être compatibles au niveau électrique avec les dispositifs TTL. On veut dire par là que les circuits intégrés 74HCT, contrairement aux circuits intégrés 74C et 74HC, peuvent être connectés directement à des éléments TTL sans interfaçage.

TABLEAU 4-2 Diverses séries de la famille de circuits logiques CMOS

Série CMOS	Préfixe	Exemple de CI
CMOS à porte métallique	40 ou 140	4001 ou 14001 (quatre portes NI)
Porte métallique, brochage compatible TTL	74C	74C02 (quatre portes NI)
Porte de silicium, brochage compatible TTL, rapide	74HC	74HC02 (quatre portes NI)
Porte de silicium, rapide, électriquement compatible à TTL	74HCT	74HCT02 (quatre portes NI)

Alimentation et masse Pour utiliser des CI, il est indispensable de raccorder correctement les broches des boîtiers. Les connexions les plus importantes sont *l'alimentation continue* et *la masse*. Ces raccordements sont nécessaires pour assurer le bon fonctionnement des circuits. Reportez-vous à la figure 4-29 où vous pouvez observer un circuit TTL et un circuit CMOS, tous deux ont une broche à la tension d'alimentation et une autre à la masse. La broche d'alimentation du circuit TTL est appelée V_{CC}, celle du circuit CMOS, V_{DD}. Nombre de nouveaux circuits intégrés CMOS, conçus pour être compatibles avec les circuits intégrés TTL, appellent aussi leur broche d'alimentation V_{CC}.

Si le CI n'est pas raccordé à l'alimentation ou à la masse, les portes logiques de la puce réagissent de façon imprévisible aux entrées logiques, de sorte qu'on n'obtient pas les niveaux logiques espérés.

Intervalles de tension des niveaux logiques Pour les éléments TTL, V_{CC} est à la valeur nominale +5 V. Pour les circuits intégrés CMOS, V_{DD} se situe entre +3 et +18 V, même si +5 V se rencontre plus souvent quand les CMOS sont utilisés dans le même système que des circuits intégrés TTL.

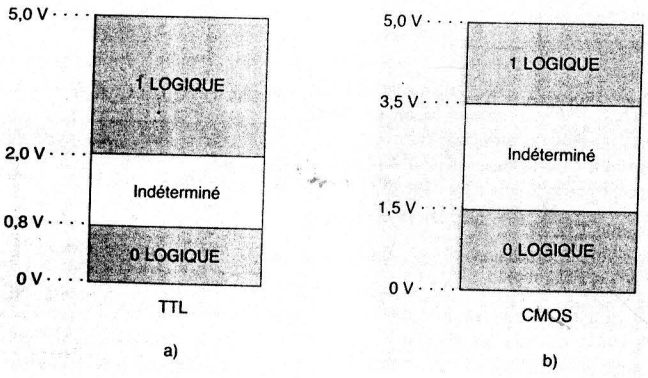
Pour les éléments TTL standard, les intervalles de tension acceptables pour les niveaux logiques 0 et 1 sont définis sur la figure 4-30 a). Ainsi, le niveau 0 est toute tension se situant entre 0 et 0,8 V; le niveau 1, toute tension se situant dans la plage de 2 V à 5 V. Les tensions qui n'apparaissent ni dans un intervalle ni dans l'autre sont dites **indéterminées** et ne devraient jamais se produire dans un élément TTL. Les fabricants de CI ne sont pas en mesure de garantir qu'un circuit TTL réagira à un signal qui se situe dans l'intervalle indéterminé (0,8 à 2,0 V).

Les intervalles des niveaux logiques pour les circuits intégrés CMOS dont le V_{DD} = +5 V sont illustrés à la figure 4-30 b). Les tensions entre 0 et 1,5 V définissent un intervalle pour le 0 logique, les tensions entre 3,5 et 5 V définissent un intervalle pour le 1 logique. La plage indéterminée va de 1,5 à 3,5 V.

Entrées non connectées (flottantes) Que se produit-il quand une entrée d'un CI numérique n'est pas raccordée? On appelle souvent « flottante » une entrée non connectée. La réponse à la question précédente varie selon qu'il s'agit d'éléments TTL ou CMOS.

Une entrée TTL flottante se comporte exactement comme s'il y avait un 1 logique. Autrement dit, la réponse du CI est celle qu'il aurait si un niveau HAUT était appliqué à

FIGURE 4-30 Intervalles de tension des niveaux logiques pour les CI numériques TTL et CMOS.



cette entrée. Cette caractéristique est souvent exploitée lors du test d'un circuit TTL. Un technicien négligent pourrait laisser certaines entrées non connectées plutôt que de les brancher sur une tension HAUTE. Bien que cela soit justifiable du point de vue logique, c'est une pratique non recommandable, particulièrement pour la conception définitive du circuit, car une entrée TTL flottante se prête particulièrement bien à la capture du bruit qui peut brouiller le fonctionnement du dispositif.

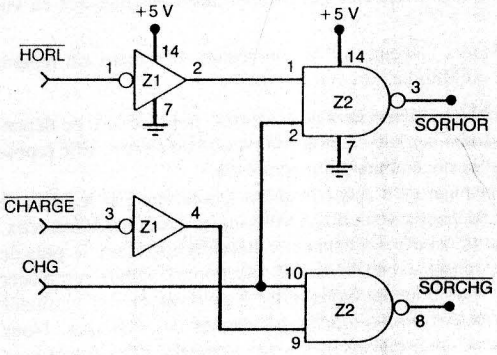
Une entrée flottante est à une tension entre 1,4 et 1,8 V quand on la vérifie avec un voltmètre ou un oscilloscope. Bien que cette tension corresponde à la situation indéterminée pour les TTL, elle donne lieu à une réponse comme si elle était un 1 logique. Retenez toujours ce comportement d'une entrée flottante, cela pourra vous rendre de grands services durant les dépannages.

Si une entrée CMOS est laissée flottante, les résultats peuvent être désastreux. Le CI peut surchauffer et possiblement se consummer de lui-même. Pour cette raison, il faut obligatoirement que toutes les entrées d'un circuit intégré CMOS soient connectées à un niveau HAUT ou un niveau BAS, ou à la sortie d'un autre CI. Une entrée CMOS flottante n'est pas à une tension c.c. précise, mais est à une tension qui fluctue de façon aléatoire car elle capte du bruit. À cause de cela, elle n'est ni à un 0 logique ni à un 1 logique: ses effets sur le signal de sortie sont imprévisibles. Parfois, la sortie oscillera en raison du bruit capté par l'entrée flottante.

Schéma de raccordement d'un circuit logique Un schéma de raccordement ou de connexion illustre *toutes* les connexions électriques, les numéros de broche, les numéros des CI, les valeurs des composants, les noms de signaux, les tensions d'alimentation. On peut voir à la figure 4-31 un schéma de raccordement type, celui d'un circuit logique simple. Examinez-le attentivement et notez les points importants que voici:

1. Ce circuit utilise des portes logiques se trouvant sur deux CI différents. Les deux INVERSEURS sont des portes de la puce 7404 dont la désignation est Z1. Le 7404 renferme six INVERSEURS; deux servent dans ce circuit et portent la désignation Z1. Dans le même ordre d'idées, les deux portes NON-ET se trouvent sur un 7400 qui renferme quatre portes NON-ET. Toutes les portes de cette puce sont désignées par la mention Z2. En numérotant les portes Z1, Z2, Z3..., nous parvenons à toujours savoir à quelle puce appartient telle porte. C'est un renseignement particulièrement précieux dans le cas des circuits complexes formés de nombreux CI comprenant plusieurs portes par puce.

FIGURE 4-31 Schéma de raccordement type d'un circuit logique.



Ic	Type
Z1	7404 six inverseurs
Z2	7400 quatre NON-ET

2. Chaque numéro de broche d'une entrée et d'une sortie de porte est indiqué sur le schéma. Ces numéros de broche et les désignations des CI permettent d'indiquer tout point d'un circuit. Ainsi on préférera se référer à la broche Z1-2 plutôt que de dire la broche de sortie de l'INVERSEUR du haut. De même, nous dirons que la broche Z1-4 est connectée à la broche Z2-9.
3. Les raccordements à l'alimentation et à la masse de chaque CI sont illustrés sur le schéma. Ainsi, la broche Z1-14 est reliée à l'alimentation +5 V et la broche Z1-7 est raccordée à la masse. Ces raccordements alimentent *tous* les INVERSEURS (six) confinés dans la puce Z1.

Les fabricants d'appareillage électronique fournissent des schémas détaillés qui ont un format très semblable à celui de la figure 4-31. Ces schémas de raccordement sont d'une nécessité absolue pour celui qui veut réparer un circuit défectueux.

QUESTIONS DE RÉVISION

1. Quel est le boîtier de circuit intégré le plus courant?
2. Nommez les cinq catégories de CI numériques selon leur complexité.
3. *Vrai ou faux*: Une puce 7474 réalise la même fonction logique et a le même brochage qu'une 74LS74.
4. *Vrai ou faux*: Une puce 74HC74 réalise la même fonction logique et a le même brochage qu'une 7474.
5. Quelle série CMOS n'a pas un brochage compatible avec la famille TTL?
6. Quel est l'intervalle admissible pour le 0 logique d'un TTL? Pour son 1 logique?
7. Même question que 6, mais cette fois pour un CMOS alimenté par $V_{DD} = +5$ V.
8. Comment un circuit intégré TTL réagit-il à une entrée flottante?
9. Comment un circuit intégré CMOS réagit-il à une entrée flottante?

4-10 DÉPANNAGE DES SYSTÈMES NUMÉRIQUES

Pour le dépannage d'un système ou d'un circuit numérique en panne, il y a trois étapes à suivre:

1. *Détection de la défaillance*. Observez le fonctionnement du circuit ou du système et comparez vos observations avec le fonctionnement normal.
2. *Délimitation de la défaillance*. Réalisez des tests et faites des mesures en vue d'isoler la défaillance.
3. *Correction de la défaillance*. Remplacez le composant défectueux ou réparez la connexion mal faite ou éliminez un court-circuit...

Même si ces étapes vous semblent élémentaires et évidentes, la procédure de dépannage suivie est grandement conditionnée par le type et la complexité du circuit. Elle repose aussi sur les outils de dépannage et la documentation disponibles.

Une bonne technique de dépannage ne s'acquiert qu'en laboratoire par le biais de l'expérimentation et de l'exercice du dépannage sur des circuits et des systèmes défectueux. Il n'y a pas de meilleure façon pour devenir un dépanneur efficace que de faire le plus de réparations possibles. Aucun livre, aussi exhaustif soit-il, ne pourra jamais remplacer l'expérience pratique. Cependant, nous pouvons vous aider à développer des aptitudes analytiques qui sont essentielles pour parvenir à trouver les pannes des systèmes. Nous allons vous décrire les types de pannes courantes dans les systèmes construits principalement

avec des CI numériques et nous allons vous montrer comment les identifier. Ensuite, nous exposerons des études de cas type afin de bien mettre en lumière les processus analytiques qui sont appliqués durant le dépannage. En outre, vous trouverez à la fin des chapitres des problèmes qui complètent la matière déjà abordée et vous donnent la chance de déduire, au moyen des techniques analytiques enseignées, les raisons causant la défaillance de circuits numériques.

Dans toutes nos discussions sur le dépannage, nous supposerons que le technicien a toujours à sa disposition une *sonde logique* et un *oscilloscope*, étant donné que ce sont là des outils standard de n'importe quel laboratoire de techniques numériques. De plus, nous supposerons que la sonde logique est dotée d'un ou de plus d'un voyant qui indique l'une des conditions suivantes au point de mesure de la sonde (pointe de la sonde):

1. Niveau BAS constant.
2. Niveau HAUT constant.
3. Train d'impulsions répétitif.
4. Un niveau BAS passant momentanément (durée d'une impulsion) au niveau HAUT.
5. Un niveau HAUT passant momentanément (durée d'une impulsion) au niveau BAS.
6. Un circuit ouvert ou un niveau logique indéterminé.

Dans les exemples de dépannage, il nous arrivera souvent de mentionner l'état indiqué par la sonde logique à divers points d'un circuit.

4-11 DÉFAILLANCES INTERNES DES CI NUMÉRIQUES

Les défaillances internes qui sont les plus fréquentes dans les CI sont:

1. Dysfonctionnement des circuits internes.
2. Courts-circuits entre des entrées ou des sorties et l'alimentation ou la masse.
3. Entrées ou sorties en circuit ouvert.
4. Courts-circuits entre deux broches (autres que celles de la masse ou de l'alimentation).

Nous décrivons ci-après chacun de ces types de défaillance.

Dysfonctionnement des circuits internes Ce type de problème survient généralement quand un des composants internes ne fonctionne tout simplement plus ou fonctionne hors de ses spécifications. Alors, les sorties des CI fournissent des valeurs n'ayant aucun rapport avec les entrées. Il n'y a aucune façon de prédire le comportement des sorties, et cela à cause de la défaillance d'un composant interne. Voici des exemples de ce type de défaillance: court-circuit de la base et de l'émetteur du transistor Q_4 ou bien une valeur pour R_2 extrêmement grande dans l'INVERSEUR TTL de la figure 4-29 a). Ce genre de défaillance est beaucoup plus rare que les trois autres.

Entrées court-circuitées à la masse ou à l'alimentation par l'intérieur Ce type de défaillance interne a pour effet de « geler » l'entrée à l'état HAUT ou l'état BAS. La figure 4-32 a) illustre la broche d'entrée 2 de la porte NON-ET d'un CI court-circuitée à la masse. À cause de cela, la broche 2 demeure toujours à l'état BAS. Si l'entrée est pilotée par un signal logique B, alors effectivement B est court-circuité à la masse. Par

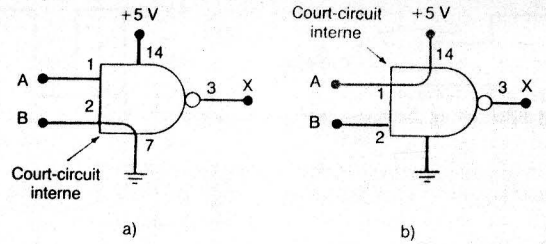


FIGURE 4-32 a) Une entrée de CI court-circuitée à la masse par l'intérieur; b) une entrée de CI court-circuitée à l'alimentation par l'intérieur. Ce type de défaillance impose à la broche court-circuitée un signal d'entrée toujours dans le même état.

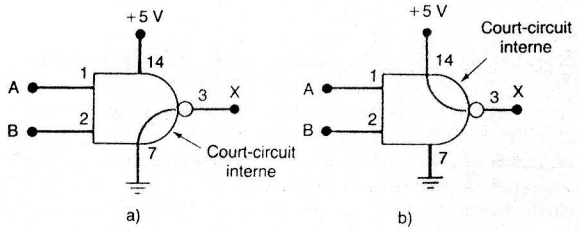


FIGURE 4-33 a) La sortie d'un CI court-circuitée à la masse par l'intérieur; b) la sortie d'un CI court-circuitée à l'alimentation par l'intérieur. Ce type de défaillance ne change pas les valeurs des signaux sur les entrées.

conséquent, une défaillance de ce type a un impact sur le dispositif qui génère le signal *B*.

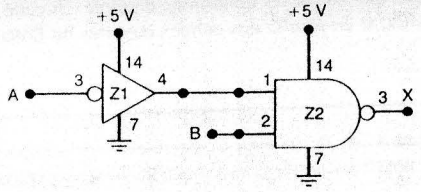
Également, une broche d'entrée d'un CI peut être court-circuitée à l'alimentation par l'intérieur, comme l'illustre la figure 4-32 b). Cela a pour effet de « geler » l'entrée à un niveau HAUT. Si la broche d'entrée est pilotée par un signal logique *A*, ce signal est effectivement porté à la tension +5 V.

Sorties court-circuitées à la masse ou à l'alimentation par l'intérieur Ce type de défaillance interne a pour effet de « geler » la broche de sortie à l'état HAUT ou à l'état BAS. La figure 4-33 a) montre la broche 3 de la porte NON-ET d'un CI court-circuitée à la masse. Cette sortie est contrainte de rester au niveau BAS, de sorte qu'elle ne réagit plus aux conditions appliquées aux broches d'entrée 1 et 2. Autrement dit, les entrées logiques *A* et *B* n'ont plus d'effet sur la sortie *X*.

Une broche de sortie de CI peut également être court-circuitée à +5 V par l'intérieur comme le montre la figure 4-33 b). Cela impose en tout temps à la broche de sortie 3 le niveau HAUT, quels que soient les états des broches d'entrée. Remarquez que ce type de défaillance n'a aucun effet sur les signaux logiques des entrées.

EXEMPLE 4-24

Reportez-vous à la figure 4-34. Un technicien se sert de sa sonde logique pour connaître les conditions sur diverses broches d'un CI. Les observations qu'il a faites sont consignées



Broche	Condition
Z1-3	Impulsion
Z1-4	BASSE
Z2-1	BASSE
Z2-2	HAUTE
Z2-3	HAUTE

FIGURE 4-34 Exemple 4-24.

sur la figure. Examinez ces résultats et concluez au bon ou au mauvais fonctionnement du circuit. Si ce circuit ne fonctionne pas bien, suggérez des défaillances possibles.

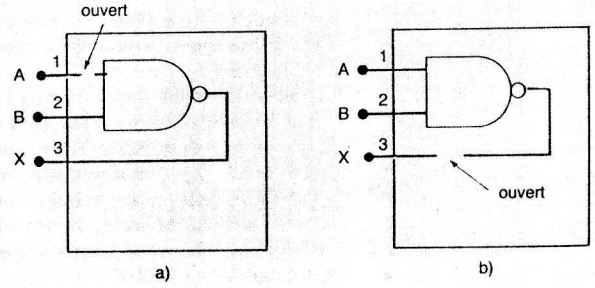
Solution La broche de sortie 4 de l'INVERSEUR devrait osciller, puisque l'entrée est un signal impulsionnel. Les mesures effectuées montrent que la broche 4 reste toujours au niveau BAS. Comme cette sortie alimente la broche 1 de Z2, la porte NON-ET est contrainte de toujours fournir une sortie HAUTE. D'après notre précédente discussion, nous pouvons énumérer trois raisons possibles expliquant un tel fonctionnement.

Première raison, il pourrait y avoir un composant défectueux dans l'INVERSEUR empêchant ce dernier de réagir correctement à l'entrée appliquée. Deuxième raison, la broche 4 de l'INVERSEUR pourrait être court-circuitée à la masse de l'intérieur, gardant ainsi la sortie de l'INVERSEUR toujours à un niveau BAS. Troisième raison, la broche 1 de Z2 pourrait être court-circuitée à la masse par l'intérieur. Cette situation empêche la sortie de l'INVERSEUR de réagir aux changements.

Entrées ou sorties en circuit ouvert Parfois, les fils de connexion très fins qui raccordent la broche d'un CI aux circuits internes peuvent se briser, ce qui ouvre le circuit. La figure 4-35 a) nous fait voir cette situation pour une broche d'entrée, et la figure 4-35 b), la même chose pour une broche de sortie.

Si une entrée d'un CI est ouverte dans le circuit, le signal logique appliqué à l'entrée n'a plus d'effet sur la sortie. Le circuit ouvert maintient l'entrée dans un état flottant. Comme nous l'avons déjà mentionné, selon que le composant est TTL ou CMOS, la réaction diffère. Les circuits intégrés TTL réagiront à une entrée ouverte en fournissant un 1 logique; les circuits CMOS réagiront aléatoirement et ils risquent d'être endommagés par surchauffe.

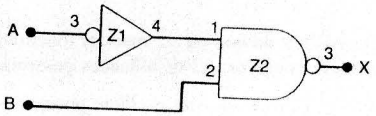
FIGURE 4-35 a) CI ayant une entrée mise en circuit ouvert par l'intérieur qui ne réagit pas aux signaux appliqués sur sa borne; b) sortie mise en circuit ouvert par l'intérieur qui ne fournit aucune tension.



Quand la sortie d'un CI est en circuit ouvert par l'intérieur, aucune tension n'est perceptible à la sortie et cela quelles que soient les conditions d'entrée. Si cette sortie est raccordée aux entrées de plusieurs CI, il en résulte des entrées ouvertes ou flottantes pour tous ces CI.

EXEMPLE 4-25

Reportez-vous au circuit de la figure 4-36, de même qu'aux relevés des observations de la sonde. Indiquez les défaillances possibles pouvant expliquer les observations obtenues. Supposez que les CI sont des TTL.



Note: Connexions V_{cc} et masse de chaque CI non montrées

Broche	Condition
Z1-3	HAUTE
Z1-4	BASSE
Z2-1	BASSE
Z2-2	Impulsion
Z2-3	Impulsion

FIGURE 4-36 Exemple 4-25.

Solution L'examen des observations relevées montre que l'INVERSEUR semble fonctionner correctement, mais que la sortie de la porte NON-ET ne correspond pas aux entrées. En effet, la sortie de la porte NON-ET devrait être HAUTE, puisque le niveau sur la broche 1 est BAS. Ce niveau BAS devrait empêcher la porte NON-ET de réagir aux impulsions appliquées sur la broche 2. Il est fort probable que le niveau BAS ne parvienne pas jusqu'aux circuits internes de la porte NON-ET en raison d'un circuit ouvert interne. Comme il s'agit d'un CI TTL, ce circuit ouvert agir exactement comme un 1 logique sur la broche 1. Si ce circuit intégré était de la famille CMOS, le circuit ouvert interne englobant la broche 1 produirait une sortie indéterminée avec possiblement risque de surchauffe et de destruction de la puce.

À la lumière de ce que nous avons déjà dit au sujet des entrées TTL ouvertes, on devrait s'attendre à ce que la tension sur la broche 1 de Z2 soit entre 1,4 et 1,8 V, et à la sonde on aurait dû observer une tension indéterminée. Ce qui aurait été le cas si le circuit ouvert avait été *externe* à la puce NON-ET. Il n'y a pas de circuit ouvert entre la broche 4 de Z1 et la broche 1 de Z2; c'est donc dire que la tension de la broche Z1-4 atteint la broche Z2-1. La déconnexion intervient à *l'intérieur* de la puce NON-ET.

Courts-circuits entre deux broches Un court-circuit interne entre deux broches d'un CI contraint les signaux logiques sur ces broches à être toujours identiques. Quand deux signaux, qui doivent être différents, varient logiquement toujours de la même façon, il y a fort à parier que ces signaux sont réunis par un court-circuit.

Analysons le circuit de la figure 4-37, dans lequel les broches 5 et 6 d'une porte NI sont court-circuitées à l'intérieur de la puce. Ce court-circuit réunit les deux sorties des INVERSEURS, ce qui fait en sorte d'avoir sur la broche Z1-2 et la broche Z1-4 le même signal malgré les variations aux entrées des INVERSEURS. À titre d'illustration, considérons les formes d'ondes d'entrée illustrées sur le schéma. Même si ces formes d'ondes sont différentes, celles sur les sorties Z1-2 et Z1-4 sont identiques. Durant l'intervalle t_1 à t_2, les deux INVERSEURS sont attaqués par un niveau HAUT sur chaque entrée et

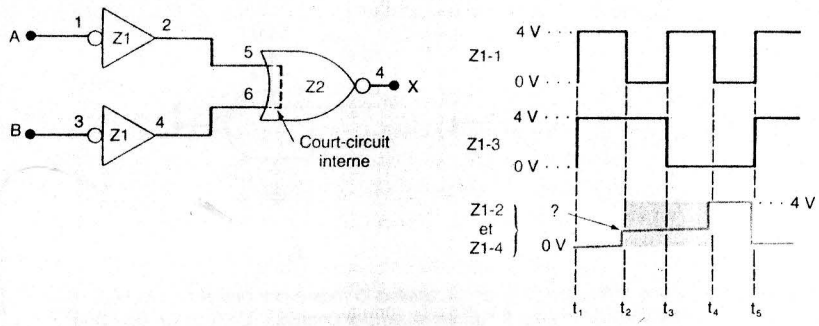


FIGURE 4-37 Quand deux broches de sortie sont court-circuitées par l'intérieur, les signaux pilotant ces broches sont contraints d'être identiques; généralement on observe un signal à trois paliers distincts.

cherchent à fournir une sortie BASSE; même si les Z1 sont court-circuités, la sortie est correcte. Durant l'intervalle t_4 à t_5, le deux INVERSEURS ont comme entrée un niveau BAS et cherchent à fournir une sortie HAUTE; encore ici, même si les Z1 sont court-circuités, la sortie est correcte. Cependant, durant les intervalles t_2 à t_3 et t_3 à t_4, un INVERSEUR essaie de produire un niveau HAUT et l'autre, un niveau BAS. Dans cette situation, le niveau réel de tension que l'on mesure sur les sorties court-circuitées dépend des circuits internes du CI. Dans le cas des dispositifs TTL, généralement cette tension sera près de la borne supérieure de l'intervalle du 0 logique (autour de 0,8 V), bien qu'elle puisse se trouver aussi dans l'intervalle indéterminée. Pour les dispositifs CMOS, le plus souvent la tension sera dans la plage indéterminée.

Chaque fois que vous observez des formes d'ondes comme le signal Z1-1 et Z1-4 de la figure 4-37, avec trois paliers distincts, vous devez suspecter deux signaux de sortie court-circuités par l'intérieur.

QUESTIONS DE RÉVISION

1. Énumérez les défaillances internes des CI.
2. Quelle défaillance interne d'un CI donne lieu à des signaux ayant trois paliers de tension distincts?

4-12 DÉFAILLANCES EXTERNES

Nous venons de voir comment identifier les effets des diverses défaillances pouvant se manifester à l'intérieur de CI numériques. Beaucoup de problèmes extérieurs aux CI peuvent aussi se produire; nous nous proposons de décrire dans la présente section les plus courants.

Lignes de signaux ouvertes Dans cette catégorie de problèmes nous englobons les défaillances qui causent une rupture ou une discontinuité dans un chemin de conduction

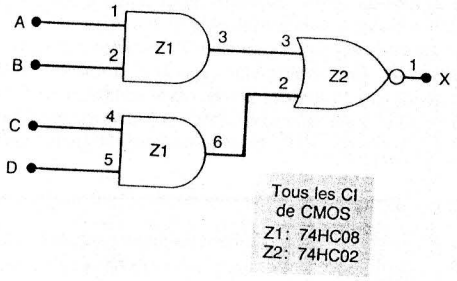
qui empêche une tension ou un signal d'aller d'un point à un autre. Voici certaines causes provoquant l'ouverture de lignes de signaux:

1. Fil brisé.
2. Mauvaise soudure de connexion; connexion enroulée lâche.
3. Fissure ou trace de coupure sur le circuit imprimé (certaines de ces ruptures sont minces comme des cheveux et on ne peut les repérer qu'avec une loupe).
4. Broche de CI brisée ou tordue.
5. Support de CI défectueux, le contact entre les broches et les prises du support n'est pas conducteur.

Ce genre de défaillance dans un circuit se diagnostique facilement; il suffit d'enlever la tension et de mesurer avec un ohmmètre la continuité entre les deux points suspectés.

EXEMPLE 4-26

Examinons le circuit CMOS de la figure 4-38 de même que les indications obtenues de la sonde logique. Quelle est la défaillance la plus probable de ce circuit?



Broche	Condition
Z1-1	Impulsion
Z1-2	BASSE
Z1-3	Impulsion
Z1-4	BASSE
Z1-5	Impulsion
Z1-6	BASSE
Z2-3	Impulsion
Z2-2	Indéterminé
Z2-1	Indéterminé

FIGURE 4-38 Exemple 4-26.

Solution Le niveau indéterminé de la sortie de la porte NI est due presque certainement à une valeur indéterminée sur la broche d'entrée 2. Comme la valeur sur Z1-6 est BASSE, ce niveau bas devrait nécessairement se retrouver sur Z2-2. De toute évidence, le niveau bas sur Z1-6 n'arrive pas jusqu'à Z2-2, et on doit suspecter un chemin de signal ouvert entre ces deux points. L'endroit où le circuit s'ouvre peut être déterminé avec la sonde logique en partant de Z1-6 en direction de Z2-2; il suffit de relever l'endroit où le signal passe de BAS à indéterminé.

Lignes de signaux court-circuitées Ce type de défaillance produit sensiblement le même effet qu'un court-circuit interne entre des broches de CI. Il contraint les deux signaux à avoir la même valeur. Ses principales causes sont:

1. *Câblage négligent.* Un exemple de ceci est le dénudage sur une trop grande longueur de fils placés à proximité.
2. *Ponts de soudure.* On désigne sous ce nom des éclaboussures de soudure qui court-circuitent deux ou plusieurs points. On observe souvent cela entre points très rapprochés, comme les broches voisines d'une puce.
3. *Attaque incomplète.* Le cuivre séparant deux chemins de conduction voisins sur un circuit imprimé n'a pas été retiré complètement par l'attaque chimique.

Un contrôle à l'ohmmètre permet aussi de constater que deux lignes de signaux sont en court-circuit.

Alimentation défectueuse Tous les systèmes numériques contiennent une ou plusieurs alimentations qui délivrent les tensions V_{CC} et V_{DD} nécessairent au fonctionnement des puces. Une alimentation défectueuse ou surchargée (qui fournit plus de courant que son intensité nominale) ne parvient pas à redresser correctement les tensions appliquées aux CI, de sorte que ces derniers fonctionnent soit mal, soit de façon imprévisible.

Une alimentation fournit une tension mal redressée quand certains de ses circuits intenses sont défectueux, ou quand les circuits qu'elle alimente drainent plus de courant qu'elle ne peut en fournir. Comme ça peut être le cas pour une puce ou un composant en panne qui soutire plus de courant que le courant normal.

Une bonne habitude de dépannage consiste donc à vérifier les niveaux de tension de l'alimentation du système en vue d'établir qu'ils se situent dans les intervalles prescris. Une autre bonne habitude est d'utiliser un oscilloscope pour déterminer les oscillations c.a. superposées aux niveaux continus et pour vérifier que le redressement des tensions reste constant durant le fonctionnement du système.

Une des indications les plus courantes d'une alimentation défectueuse est la constatation d'un mauvais fonctionnement ou de la panne complète d'une ou plusieurs puces. Certains CI sont plus ou moins tolérants à des variations de l'alimentation, tandis que d'autres ne le sont pas du tout. Vous devriez toujours vérifier les niveaux de l'alimentation et de la masse de chaque CI qui ne semble pas fonctionner correctement.

QUESTIONS DE RÉVISION

1. Nommez les types de défaillance externe les plus courants.
2. Indiquez certaines causes à l'origine de circuits ouverts dans un chemin de signaux.
3. Quels symptômes sont imputables à une alimentation défectueuse?

4-13 ÉTUDE D'UN CAS DE DÉPANNAGE

L'exemple suivant se veut une illustration de la démarche analytique mise en œuvre dans le dépannage de circuits numériques. Même si le circuit utilisé est un circuit combinatoire relativement élémentaire, le raisonnement tenu et les procédures de dépannage appliquées conviennent aussi bien aux circuits numériques plus complexes que nous rencontrerons dans les chapitres à venir.

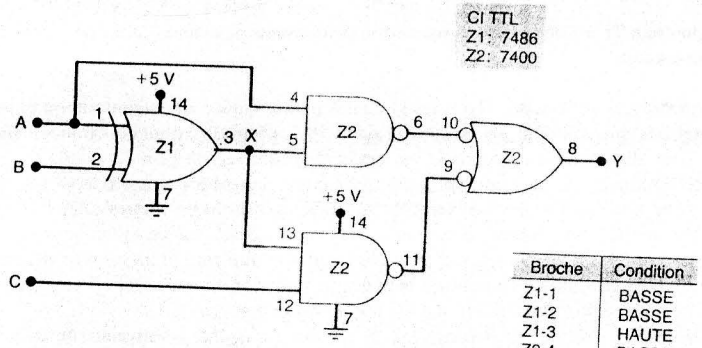
EXEMPLE 4-27

Soit le circuit de la figure 4-39. La sortie Y doit passer au niveau HAUT quand l'une ou l'autre des conditions suivantes est réalisée:

1. $A = 1, B = 0$, quel que soit C
2. $A = 0, B = 1, C = 1$

Vous pouvez vérifier cette assertion par vous-même.

Au moment du test de ce circuit, la technicienne remarque que la sortie Y passe au niveau HAUT quand A est HAUT ou C est HAUT, quel que soit le niveau de B. Elle effectue des mesures à la sonde logique pour la condition $A = B = C = 1$ et relève les indications reportées sur la figure 4-39.



Broche	Condition
Z1-1	BASSE
Z1-2	BASSE
Z1-3	HAUTE
Z2-4	BASSE
Z2-5	HAUTE
Z2-6, 10	HAUTE
Z2-13	HAUTE
Z2-12	HAUTE
Z2-9, 11	BASSE
Z2-8	HAUTE

FIGURE 4-39 Exemple 4-27.

Examinez les niveaux enregistrés et donnez les causes probables du dysfonctionnement. Ensuite, élaborez une procédure pas à pas visant à déterminer la défaillance exacte.

Solution Toutes les sorties des portes NON-ET affichent le bon niveau de tension compte tenu des niveaux à leurs entrées. Au contraire, la porte OU exclusif devrait donner un niveau BAS sur la broche de sortie 3, puisque ses deux entrées sont BASSES simultanément. Il apparaît donc que Z1-3 est « geler » au niveau HAUT, même quand les entrées devraient donner un niveau BAS. Il y a plusieurs causes à cet état de chose:

1. Défaillance d'un composant interne de Z1 empêchant la transition de la sortie vers le niveau BAS.
2. Court-circuit externe avec V_{CC} d'un point quelconque des conducteurs aboutissant au nœud X.
3. Broche 3 de Z1 court-circuitée à V_{CC} par l'intérieur.
4. Broche 5 de Z2 court-circuitée à V_{CC} par l'intérieur.
5. Broche 13 de Z2 court-circuitée à V_{CC} par l'intérieur.

Toutes ces possibilités, sauf la première, impliquent un court-circuit entre le nœud X (et toute broche raccordée à ce dernier) et l'alimentation V_{CC}.

La procédure suivante peut être employée pour cerner la défaillance. Elle n'est pas unique, car, comme nous l'avons déjà mentionné, la procédure de dépannage retenue par un technicien est grandement conditionnée par l'appareillage d'essai disponible.

1. Vérifiez le niveau de V_{CC} et celui de la masse des broches appropriées de Z1. Même s'il est peu vraisemblable que l'absence des bonnes tensions sur ces broches causent le maintien de Z1-3 au niveau HAUT, c'est toujours une bonne habitude de tester ces broches dans tout CI qui fournit une sortie incorrecte.
2. Coupez l'alimentation du circuit et avec un ohmmètre recherchez un court-circuit (résistance inférieure à 1 Ω) entre le nœud X et tout point connecté à V_{CC} (par exemple Z1-14 ou Z2-14). Si vous ne trouvez pas de court-circuit, vous éliminez les quatre dernières possibilités de la liste. Cela signifie qu'il est très probable qu'il y ait un court-circuit interne dans Z1 et qu'il faille le remplacer.

3. Si à l'étape 2 vous avez trouvé un court-circuit entre le nœud X et V_{CC}, examinez dans les moindres détails la plaquette de circuit à la recherche de ponts de soudure, de traces de cuivre non enlevées, de fils non isolés se touchant ou de tout autre anomalie pouvant être à l'origine du court-circuit externe avec V_{CC}. Un endroit probable pour un pont de soudure pourrait être entre les broches adjacentes 13 et 14 de Z2. La broche 14 est raccordée à V_{CC}, tandis que la broche 13 est raccordée au nœud X. Si vous repérez un court-circuit externe, revérifiez avec l'ohmmètre qu'il n'y a plus de court-circuit entre X et V_{CC}.
4. Si à l'étape 3, vous n'avez trouvé aucun court-circuit externe, les trois possibilités restantes sont les courts-circuits internes entre V_{CC} et Z1-3, Z2-13 ou Z2-5. L'une de ces défaillances met en court-circuit le nœud X et V_{CC}.

Pour trouver lequel de ces CI est le coupable, nous devrons déconnecter chacun d'eux du nœud X *un à la fois*, et rechercher un court-circuit avec V_{CC} après chaque déconnexion. Dès que la broche qui est en court-circuit avec V_{CC} sera déconnectée, le nœud X cessera de rester à la tension de V_{CC}.

Le travail de déconnexion de chaque broche suspecte aboutissant à X peut être simple ou complexe dépendamment de la construction du circuit. Si les CI sont dans des supports, il suffit de les extraire du support, de replier la broche suspecte et de les réenficher. Par contre, si les CI sont soudés dans la plaquette de circuit imprimé, vous aurez à couper la trace qui aboutit à la broche (ou couper la broche), puis à tout réparer après.

Il existe une méthode de dépannage qui élimine tout repliement de broche ou coupure de trace quand il s'agit d'isoler un court-circuit. On a besoin dans ce cas d'un outil appelé une *sonde ampèremétrique* qui relève le courant qui parcourt le court-circuit quand la tension au nœud est impulsionnelle. La sonde ampèremétrique capte le champ magnétique variable autour du conducteur qui agit comme court-circuit pour le courant. Nous reviendrons sur cette technique au Chapitre 8.

L'exemple 4-27 malgré sa simplicité illustre le genre de raisonnement que doit pratiquer le dépanneur pour parvenir à isoler le problème. Vous aurez la possibilité de pouvoir commencer à exercer vos talents de dépanneur en résolvant les problèmes 4-34 à 4-44.

PROBLÈMES

SECTIONS 4-2 et 4-3

4-1 Simplifiez les expressions suivantes au moyen de l'algèbre booléenne.

a) $x = ABC + \overline{A}C$

b) $y = (Q + R)(\overline{Q} + \overline{R})$

c) $w = ABC + A\overline{B}C + \overline{A}$

d) $q = \overline{RST}(\overline{R} + S + T)$

e) $x = \overline{A}\overline{B}\overline{C} + \overline{A}BC + ABC + A\overline{B}\overline{C} + A\overline{B}C$

f) $z = (B + \overline{C})(\overline{B} + C) + \overline{\overline{A} + B + \overline{C}}$

g) $y = (\overline{C} + D) + \overline{A}C\overline{D} + A\overline{B}\overline{C} + \overline{A}\overline{B}CD + AC\overline{D}$

4-2 Simplifiez le circuit de la figure 4-40 au moyen de l'algèbre booléenne.

4-3 Remplacez chacune des portes de la figure 4-40 par une porte NI, puis simplifiez le circuit résultant au moyen de l'algèbre booléenne.

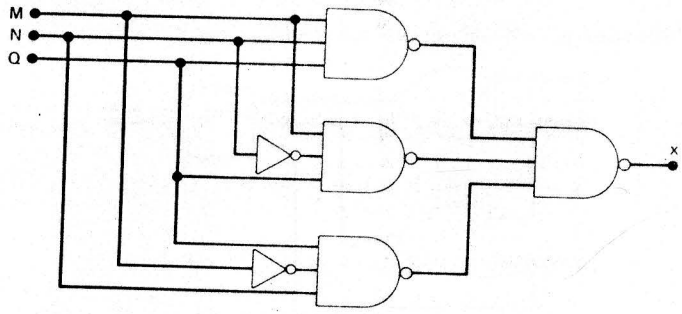


FIGURE 4-40

SECTION 4-4

4-4 Concevez un circuit logique qui correspond à la table de vérité suivante.

A	*B*	*C*	*x*
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

4-5 Concevez un circuit logique dont la sortie est au niveau HAUT *seulement* quand au moins deux des trois entrées A, B et C sont au niveau BAS.

4-6 Trois cellules photoélectriques sont éclairées par trois rayons lumineux clignotants. Ces rayons doivent clignoter de manière séquentielle pour que les trois rayons ne soient jamais allumés ou éteints en même temps. Chaque cellule sert à contrôler un de ces rayons et constitue un circuit qui produit une tension de sortie BASSE quand elle est dans le noir et une tension de sortie HAUTE quand elle est éclairée. Créez un circuit logique dont les entrées sont les tensions de sortie de ces trois cellules et qui doit produire un signal de niveau HAUT quand les trois rayons sont allumés ou éteints en même temps.

4-7 Un nombre binaire de 4 bits est noté $A_3A_2A_1A_0$, où A_3, A_2, A_1 et A_0 représentent chacun des bits, A_0 étant le bit de poids le plus faible. Concevez un circuit logique qui donne une sortie HAUTE quand le nombre binaire est supérieur à 0010 et inférieur à 1000.

4-8 La figure 4-41 illustre le schéma d'un circuit d'alarme d'une automobile qui détecte diverses situations non souhaitables. Les trois interrupteurs servent à désigner l'état de la porte du conducteur, de l'allumage et des phares, respectivement. Concevez le circuit logique ayant ces trois interrupteurs comme entrées, qui déclenche l'alarme quand l'une des situations que voici se produit:

- Les phares sont allumés et l'allumage est coupé.
- La porte est ouverte et le contact d'allumage est mis.

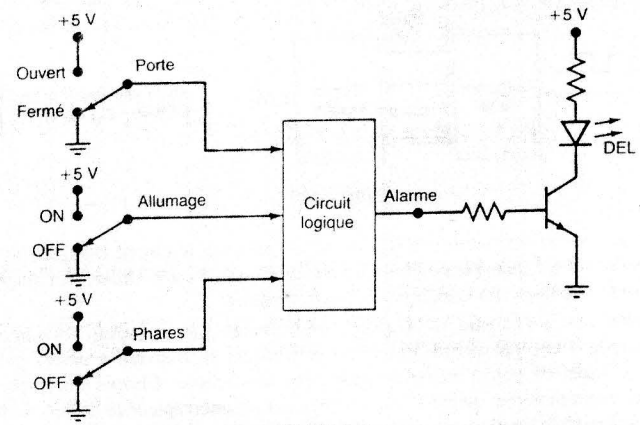


FIGURE 4-41

4-9 Réalisez le circuit du problème 4-4 en n'utilisant que des portes NON-ET.

4-10 Réalisez le circuit du problème 4-5 en n'utilisant que des portes NON-ET.

4-11 Matérialisez l'expression $z = \overline{D} + ABC + \overline{A}\overline{C}$ au moyen de portes ET, OU et NON, puis substituez à toutes ces portes des portes NON-ET.

SECTION 4-5

4-12 Simplifiez l'expression du problème 4-1 e) au moyen d'un diagramme K.

4-13 Simplifiez l'expression du problème 4-1 g) au moyen d'un diagramme K.

4-14 Simplifiez l'expression du problème 4-7 au moyen d'un diagramme K.

4-15 Trouvez les expressions minimales de chacun des diagrammes K de la figure 4-42.

	$\overline{C}\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
$\overline{A}\overline{B}$	1	1	1	1
$\overline{A}B$	1	1	0	0
AB	0	0	0	1
$A\overline{B}$	0	1	1	0

a)

	$\overline{C}\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
$\overline{A}\overline{B}$	1	0	1	1
$\overline{A}B$	1	0	0	1
AB	0	0	0	0
$A\overline{B}$	1	0	1	1

b)

	$\overline{C}$	C
$\overline{A}\overline{B}$	1	1
$\overline{A}B$	0	0
AB	1	0
$A\overline{B}$	1	X

c)

FIGURE 4-42

4-16 La figure 4-43 montre un *compteur DCB* qui fournit une sortie sur 4 bits correspondant au code DCB du nombre d'impulsions appliquées à l'entrée du compteur. Par exemple, après quatre impulsions, les broches de sortie *DCBA* du compteur portent la valeur $0100_2 = 4_{10}$. Le compteur se remet à 0000 à la dixième impulsion et reprend son comptage de zéro. Autrement dit, les sorties *DCBA* ne peuvent jamais

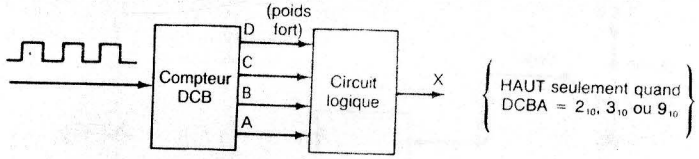


FIGURE 4-43

avoir de valeur supérieure à $1001_2 = 9_{10}$. Concevez le circuit logique qui donne un niveau HAUT quand le nombre d'impulsions est 2, 3 ou 9. Utilisez un diagramme K et tirez avantage des conditions « indifférentes ».

4-17 On peut voir à la figure 4-44 quatre interrupteurs faisant partie des circuits de commande d'une photocopieuse. Ces interrupteurs se trouvent à divers points le long du trajet du papier xérographique dans la machine. Chaque interrupteur est normalement ouvert et quand le papier passe sur un interrupteur ce dernier se ferme. Il est impossible que les interrupteurs S1 et S2 soient fermés simultanément. Concevez le circuit logique qui fournit un niveau HAUT chaque fois qu'*au moins deux interrupteurs* sont fermés en même temps. Utilisez un diagramme K et tirez avantage des conditions indifférentes.

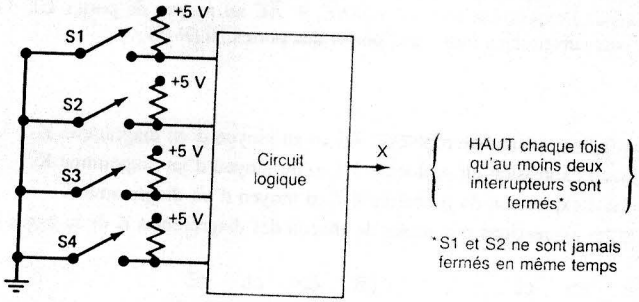


FIGURE 4-44

SECTION 4-6

4-18 Trouvez la forme d'onde de sortie du circuit de la figure 4-45.

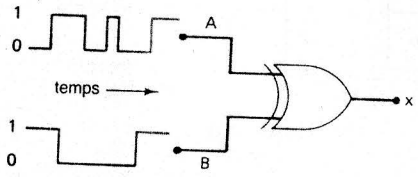


FIGURE 4-45

a) Refaites le problème mais cette fois-ci en maintenant l'entrée *B* à 0.
b) Refaites le problème mais cette fois-ci en maintenant l'entrée *B* à 1.

4-19 Déterminez les conditions d'entrée qui produisent $x = 1$ à la figure 4-46.

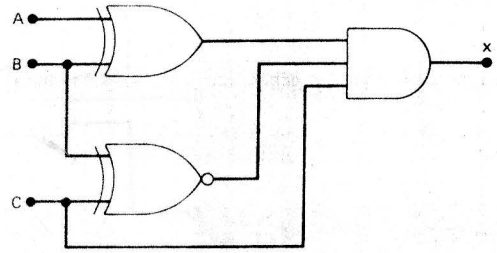


FIGURE 4-46

4-20 Refaites le problème 4-6, mais cette fois-ci en utilisant des portes OU exclusif ou NI exclusif et une autre porte.

4-21 La figure 4-47 représente un *détecteur de grandeur relative* qui accepte deux nombres binaires de 3 bits $x_2x_1x_0$ et $y_2y_1y_0$ et qui établit si ces nombres sont égaux et, dans la négative, celui qui est le plus grand. Il possède trois sorties définies ainsi:

a) $M = 1$ si les deux nombres sont égaux.
b) $N = 1$ si $x_2x_1x_0$ est supérieur à $y_2y_1y_0$.
c) $P = 1$ si $y_2y_1y_0$ est supérieur à $x_2x_1x_0$.

Concevez les circuits logiques du comparateur qui comprend *six* entrées et *trois* sorties. Ces circuits sont beaucoup trop complexes pour qu'on puisse recourir à la solution de la table de vérité. Nous vous suggérons de revoir l'exemple 4-16; vous y trouverez peut-être une façon d'aborder ce problème.

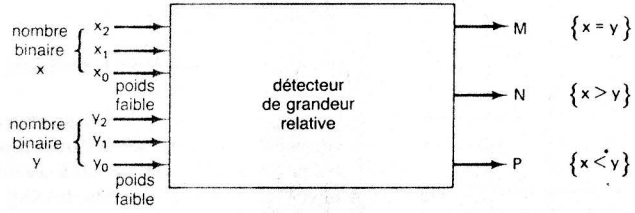
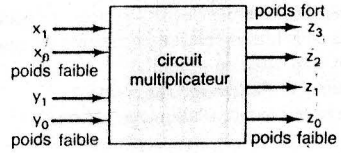


FIGURE 4-47

AUTRES PROBLÈMES DE CONCEPTION

4-22 La figure 4-48 représente un circuit multiplicateur qui accepte deux nombres binaires de 2 bits x_1x_0 et y_1y_0 et qui fournit en sortie le nombre binaire $Z_3Z_2Z_1Z_0$ égal au produit arithmétique des deux nombres d'entrée. Créez le circuit logique du multiplicateur. (*Suggestion*: Ce circuit logique doit avoir quatre entrées et quatre sorties.)

FIGURE 4-48



4-23 Un code DCB est transmis à un récepteur à distance. Les bits sont A_3, A_2, A_1, A_0, A_3 étant le bit de poids le plus fort. Les circuits du récepteur comprennent un circuit *détecteur d'erreur DCB* (c.-à-d. 1001). Concevez ce circuit pour que la sortie soit HAUTE quand il y a une erreur.

4-24 Concevez un circuit logique qui génère une sortie HAUTE quand A et B sont tous les deux à 1 et quand C et D sont ou bien tous les deux à 0 ou bien tous les deux à 1. Essayez de trouver la solution sans recourir à une table de vérité. Ensuite, dressez la table de vérité du circuit trouvé et voyez si cela corrobore l'énoncé du problème.

4-25 Quatre grandes cuves dans une usine de fabrication de produits chimiques contiennent différents liquides chauffés. Des capteurs limnimétriques (de niveau) servent à déceler le dépassement d'un niveau préétabli dans les cuves A et B. Des capteurs thermométriques surveillent la température des cuves C et D pour qu'elle ne descende pas sous une valeur de consigne. Supposez que les capteurs limnimétriques sont à 0 quand le niveau est correct et à 1 quand il est trop haut. En outre, supposez que les capteurs thermométriques sont à 0 quand la température est acceptable et à 1 quand elle est trop basse. Concevez un circuit logique qui sonne l'alerte quand se produisent en même temps un niveau trop haut dans A ou B et une température trop basse dans la cuve C ou la cuve D.

4-26 La figure 4-49 nous montre l'intersection entre une route principale et une route secondaire. Des capteurs de voitures ont été placés le long des voies C et D (route principale) et des voies A et B (route secondaire). Les sorties de ces capteurs sont à 0 quand il n'y a pas de voiture et à 1 quand il y en a. Le feu de circulation se trouvant à cette intersection est commandé par les règles de décision suivantes:

1. Le feu E-O est vert quand il y a des voitures dans les *deux* voies C et D.
2. Le feu E-O est vert quand il y a des voitures dans C *ou* D et quand il y en a dans A ou dans B mais *pas dans les deux*.
3. Le feu N-S est vert quand il y a des voitures dans les voies A *et* B et qu'il y en a dans C ou dans D mais *pas dans les deux*.
4. Le feu N-S est aussi vert quand il y a des voitures dans A *ou* B et qu'il n'y a pas de voitures dans C *et* D.
5. Le feu E-O est vert quand il n'y a *pas* de voiture *du tout*.

En utilisant les tensions de sortie des capteurs A, B, C et D comme entrées, concevez un circuit logique qui commande le feu de circulation. Ce circuit a deux sorties, soit E-O et N-S, qui prennent la valeur HAUTE quand le feu doit être *vert*. Simplifiez le plus possible ce circuit et illustrez *toutes* les étapes.

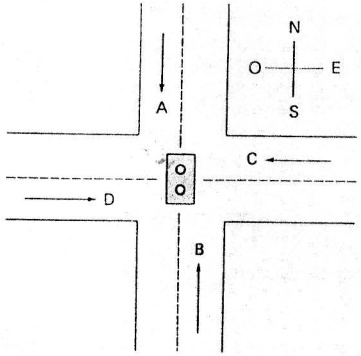


FIGURE 4-49

SECTION 4-7

4-27 Reconcevez le générateur et contrôleur de parité de la figure 4-24 pour qu'il utilise la parité impaire (*Indication*: quelle est la relation entre le bit de parité paire et le bit de parité impaire pour le même ensemble de données binaires?)

4-28 Reconcevez le générateur et le contrôleur de parité de la figure 4-24 pour qu'il fonctionne avec des données de 8 bits.

SECTION 4-8

4-29 Construisez un circuit logique qui autorise le passage d'un signal d'entrée jusqu'à la sortie quand une entrée de commande B est BASSE et qu'au même moment une entrée de commande C est HAUTE; dans tous les autres cas la sortie reste à 0.

4-30 Construisez un circuit qui *bloque* le passage d'un signal d'entrée seulement quand les entrées de commande B, C et D sont toutes HAUTES; la sortie de ce circuit est à 1 dans la situation de blocage.

4-31 Concevez un circuit logique qui commande le passage d'un signal A lorsque:

1. La sortie X est identique au signal A quand les entrées de commande B et C ont la même valeur.
2. X reste au niveau HAUT quand B et C sont différents.

4-32 Concevez un circuit logique qui a deux signaux d'entrée A_1 et A_0 et une entrée de commande S et dont le fonctionnement respecte les exigences exprimées sur la figure 4-50. Ce type de circuit est appelé un *multiplexeur* (voir chapitre 9).

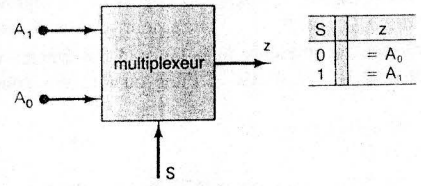


FIGURE 4-50

4-33 Utilisez la méthode des diagrammes de Karnaugh pour concevoir un circuit qui respecte les exigences exprimées à l'exemple 4-16. Comparez votre solution avec celle de la figure 4-22. Le présent exercice démontre que les diagrammes de Karnaugh ne peuvent tirer avantage de la logique des portes OU exclusif et NI exclusif. Le concepteur doit être en mesure de déterminer à quels moments il lui faut utiliser ces portes.

SECTIONS 4-9 à 4-13

* **4-34** Les signaux illustrés sur la figure 4-51 sont appliqués aux entrées du circuit de la figure 4-31. Supposons qu'il y ait un circuit ouvert interne en Z1-4.

a) Quel est l'indication de la sonde en Z1-4?
b) Quelle tension continue devrait-on relever sur un voltmètre en Z1-4? (N'oubliez pas que tous les CI sont des TTL).
c) Tracez la forme que devraient avoir les signaux $\overline{SORHOR}$ et $\overline{SORCHG}$.

* Dans l'ensemble de ce livre, la sonde logique indique un problème de dépannage.

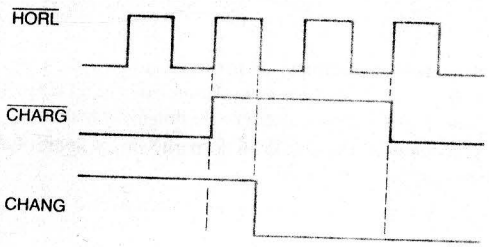


FIGURE 4-51

4-35 Les signaux de la figure 4-51 sont appliqués aux entrées du circuit de la figure 4-31. Supposons qu'il y ait un court-circuit interne entre les broches 9 et 10 de Z2. Tracez la forme du signal probable en Z2-10, $\overline{SORHOR}$ et $\overline{SORCHG}$.

4-36 Supposons que les CI de la figure 4-31 soient des CMOS. Décrivez la perturbation que subit le fonctionnement de ce circuit si un circuit ouvert existe dans le conducteur entre Z2-2 et Z2-10.

4-37 Dans l'exemple 4-24, nous avons énuméré trois défaillances possibles pouvant expliquer la situation de la figure 4-34. Quelle procédure devrions-nous suivre pour déterminer la vraie défaillance parmi les trois?

4-38 Reportez-vous au circuit de la figure 4-36. Supposons que les dispositifs soient des éléments CMOS. En outre, supposons que la sonde logique indique que la tension sur Z2-3 est « indéterminée » plutôt que « pulsatoire ». Énumérez les défaillances possibles et rédigez la procédure à suivre pour identifier la défaillance en cause.

4-39 Reportez-vous au circuit logique de la figure 4-39. Rappelez-vous que la sortie Y doit prendre le niveau HAUT quand l'une ou l'autre des conditions suivantes se réalise:

1. $A = 1, B = 0$, quel que soit C
2. $A = 0, B = 1, C = 1$

Au moment de la vérification du circuit, le technicien observe que Y passe au niveau HAUT seulement dans le cas de la première condition, mais reste BAS quand l'autre condition d'entrée se réalise. Examinez la liste suivante des défaillances possibles. Pour chaque hypothèse, inscrivez un « oui » ou un « non », selon que vous considérez que c'est la bonne réponse ou non. Pour chaque non donné, justifiez en indiquant le raisonnement suivi.

a) Un court-circuit interne à la masse en Z2-13.
b) Un circuit ouvert dans la connexion aboutissant à Z2-13.
c) Un court-circuit interne à V_{CC} en Z2-11.
d) Un circuit ouvert dans la connexion entre V_{CC} et Z2.
e) Un circuit ouvert interne en Z2-19.
f) Un circuit ouvert dans la connexion Z2-11 et Z2-9.
g) Un pont de soudure entre les broches 6 et 7 de Z2.

4-40 Élaborez une procédure visant à isoler la défaillance à l'origine du dysfonctionnement décrit dans le problème 4-39.

4-41 Supposons que les portes de la figure 4-39 sont toutes des CMOS. Quand un technicien vérifie ce circuit, il constate que le circuit fonctionne correctement sauf pour les conditions suivantes:

1. $A = 1, B = 0, C = 0$
2. $A = 0, B = 1, C = 1$

Pour ces conditions, la sonde logique indique des niveaux indéterminés en Z2-6, Z2-11 et Z2-8. D'après vous, quelle est la panne la plus plausible de ce circuit? Développez votre raisonnement.

4-42 La figure 4-52 représente un circuit logique combinatoire qui actionne un signal d'alarme dans une auto quand le siège du conducteur et/ou du passager sont occupés et que les ceintures de sécurité ne sont pas attachées au moment où l'auto est mise en route. Les signaux vrais au niveau haut *COND* et *PASS* signalent la présence, respectivement, du conducteur et du passager, et ils sont fournis par des capteurs de pression logés dans les sièges. Le signal *ALLU* est vrai au niveau HAUT quand le contact d'allumage est mis. Le signal $\overline{CEINC}$ est vrai au niveau BAS et informe que la ceinture du conducteur n'est *pas attachée*; $\overline{CEINP}$ joue le même rôle pour la ceinture du passager. Le signal d'alarme se déclenche (BAS) si l'auto démarre et que l'un ou les deux occupants avant n'ont pas attaché leur ceinture.

a) Vérifiez que ce circuit a bien le comportement décrit.
b) Décrivez le fonctionnement de ce système d'alarme si Z1-2 est court-circuité à la masse par l'intérieur.
c) Décrivez son fonctionnement s'il existe une connexion ouverte entre Z2-6 et Z2-10.

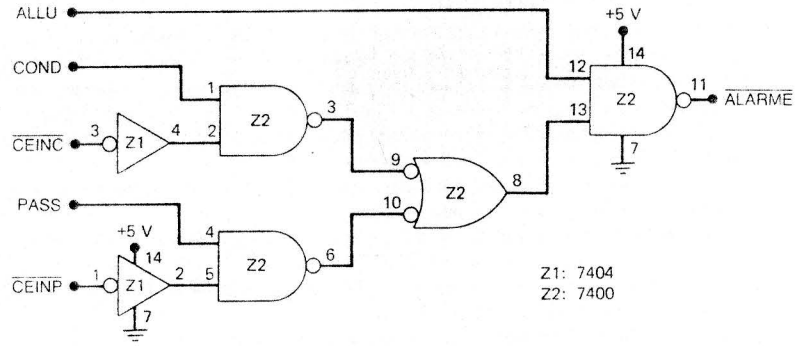


FIGURE 4-52

4-43 Supposons que le système de la figure 4-52 se met en marche dès que le conducteur ou son passager sont assis dans l'auto et que celle-ci est mise en route, que l'on ait attaché ou non les ceintures. Quelles sont les défaillances possibles? Quelle procédure appliquerez-vous pour identifier la véritable défaillance qui s'est produite?

4-44 Supposons que le système d'alarme de la figure 4-52 se comporte de telle manière qu'une alarme retentit dès qu'on fait démarrer l'auto quel que soit l'état des autres entrées. Donnez la liste des défaillances possibles et rédigez une procédure visant à isoler la défaillance.

CONTRÔLE

4-45 Définissez chacunes des expressions suivantes:

a) diagramme de Karnaugh
b) forme d'une somme de produits
c) générateur de parité
d) octet
e) circuit de blocage

f) état indifférent
g) entrée flottante
h) niveau de tension indéterminé

APPLICATIONS AU MICRO-ORDINATEUR

4-46 Dans un micro-ordinateur, le microprocesseur (MP) est toujours en communication avec l'une des ressources suivantes: 1) de la mémoire vive (RAM) qui conserve les programmes et les données susceptibles de modification; 2) de la mémoire morte (ROM) qui conservent les programmes et les données non modifiables; 3) des dispositifs d'entrée/sortie (E/S) comme le clavier, les écrans, les imprimantes, les lecteurs de disque. Pendant qu'il exécute le programme, le MP génère un code d'adresse dont le rôle est de choisir la ressource (VIVE, MORTE ou E/S) avec laquelle le MP veut communiquer. La figure 4-53 nous illustre un montage type dans lequel le microprocesseur sort sur 8 bits désignés A_{15} à A_8 un code d'adresse. Dans la réalité, les MP sortent des codes d'adresse de 16 (ou 32) bits, dans lesquels les bits de poids faible A_7 à A_0 n'interviennent pas dans la sélection du dispositif. Le code d'adresse est appliqué à un circuit logique qui décode l'adresse pour produire les signaux de sélection de dispositif — $\overline{VIVE}$, $\overline{MORTE}$ et $\overline{E/S}$.

Analysez ce circuit et trouvez ce qui suit:

a) L'intervalle des adresses A_{15} à A_8 qui activent la broche $\overline{VIVE}$.
b) L'intervalle des adresses qui activent $\overline{E/S}$.
c) L'intervalle des adresses qui activent $\overline{MORTE}$.

Exprimez les adresses en binaire et en hexadécimal. Par exemple, la réponse de la question a) est A_{15} à A_8 = 00000000_2 à 11101111_2 = 00_{16} à EF_{16}.

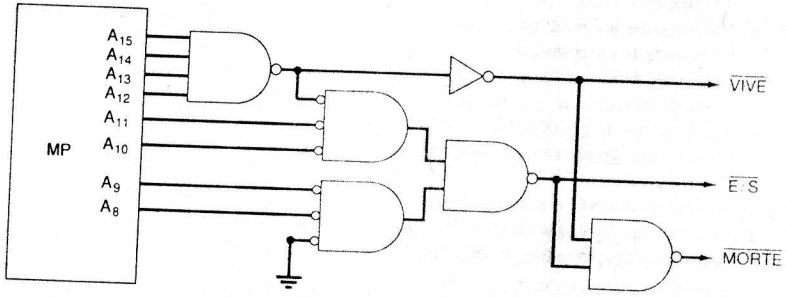


FIGURE 4-53

4-47 Dans certains micro-ordinateurs, le MP peut être *invalidé* pour de courtes périodes de temps pendant qu'un autre dispositif contrôle la mémoire VIVE, la mémoire MORTE et les E/S. Lors de ces périodes, un signal spécial ($\overline{DMA}$) est activé par le MP en vue d'invalider (désactiver) le circuit logique de sélection de dispositif et de conserver $\overline{VIVE}$, $\overline{MORTE}$, et $\overline{E/S}$ dans leur état inactivé. Modifiez le circuit de la figure 4-53 pour que $\overline{VIVE}$, $\overline{MORTE}$, et $\overline{E/S}$ soient désactivés chaque fois que $\overline{DMA}$ est actif, quel que soit l'état du code d'adresse.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 4-3

1. L'expression b) n'est pas sous forme de somme de produits, à cause du signe d'inversion au-dessus des variables C et D (par ex. le terme $\overline{ABD}$). L'expression c) n'est pas sous forme d'une somme de produits, à cause du terme $(M + \overline{N})P$ **3.** $x = \overline{A} + \overline{B} + \overline{C}$.

SECTION 4-4

1. $x = \overline{A}\overline{B}\overline{C}D + \overline{A}B\overline{C}\overline{D} + \overline{A}B\overline{C}D$. **2.** huit

SECTION 4-5

1. $x = AB + AC + BC$ **2.** $x = A + BCD$ **3.** Une condition pour laquelle il n'y a pas de condition de sortie précise.

SECTION 4-6

Un niveau BAS constant. **2.** Non, la porte OU exclusive disponible peut servir comme INVERSEUR si on raccorde une de ses entrées à un HAUT constant (voir exemple 4-15).

SECTION 4-8

1. $x = \overline{A(B \oplus C)}$ **2.** OU, NON-ET **3.** NON-ET, NI

SECTION 4-9

1. DIP **2.** SSI, MSI, LSI, VLSI, ULSI. **3.** Vrai. **4.** Vrai. **5.** Série 40 ou 140. **6.** 0 V à 0,8 V; 2,0 à 5,0 V. **7.** 0 V à 1,5 V; 3,5 V à 5 V. **8.** Comme si l'entrée était un niveau HAUT. **9.** Imprévisible; peut surchauffer et être détruit.

SECTION 4-11

1. Entrées ou sorties en circuit ouvert; entrées ou sorties court-circuitées à V_{CC}; entrées ou sorties court-circuitées à la masse; broches court-circuitées; défaillances internes du circuit. **2.** Broches court-circuitées

SECTION 4-12

1. Lignes de signaux en circuit ouvert; lignes de signaux en court-circuit; alimentations défectueuses. **2.** Fils coupés, connexions soudées négligentes; fissures ou coupures de la plaquette de circuit imprimé; broches brisées ou courbées; supports de CI défectueux. **3.** Des CI fonctionnant de façon imprévisible ou ne fonctionnant pas du tout.

,n instant
, antérieure

5

BASCULES ET ÉLÉMENTS CONNEXES

Contenu

5-1 Élément de mémoire en portes NON-ET
5-2 Élément de mémoire en portes NI
5-3 Étude d'un cas de dépannage
5-4 Signal d'horloge et bascules synchrones
5-5 Bascule S-C synchrone
5-6 Bascule J-K synchrone
5-7 Bascule D synchrone
5-8 Élément de mémoire D
5-9 Entrées asynchrones
5-10 Symboles IEEE/ANSI
5-11 Considérations sur la synchronisation des bascules
5-12 Problème de synchronisation possible dans les circuits à bascules
5-13 Bascule maître/esclave
5-14 Applications des bascules
5-15 Synchronisation des bascules
5-16 Détection d'une séquence d'entrée
5-17 Stockage et transfert de données
5-18 Transfert en série: registres à décalage
5-19 Division de la fréquence et comptage
5-20 Application au micro-ordinateur
5-21 Dispositifs à bascule de Schmitt
5-22 Monostable
5-23 Minuterie CI 555
5-24 Minuterie 555 vue comme un monostable
5-25 Analyse de circuits séquentiels
5-26 Multivibrateurs astables
5-27 Dépannage de circuits à bascules
5-28 Récapitulation des bascules

Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Construire un élément de mémoire en portes NON-ET et en portes NI et en analyser le fonctionnement.
- Éliminer les rebonds d'un interrupteur mécanique au moyen d'un circuit à mémoire.
- Indiquer la différence entre systèmes synchrones et systèmes asynchrones.
- Comprendre les divers types de bascules déclenchées par un signal, comme les J-K, les D et les S-C.
- Analyser et mettre en pratique les divers paramètres de synchronisation spécifiés par les fabricants.
- Décrire un circuit d'aiguillage d'impulsion et de détection de front.
- Comprendre les principales différences entre les transferts de données en série et les transferts en parallèle.
- Tracer les formes d'ondes de synchronisation apparaissant à la sortie de plusieurs types de bascules en réponse à un ensemble de signaux d'entrée.
- Analyser les divers symboles IEEE/ANSI pour les bascules.
- Utiliser des graphes de transition d'états pour décrire le fonctionnement d'un compteur.
- Énumérer un certain nombre d'applications des bascules.
- Exploiter les bascules dans les circuits de synchronisation.
- Raccorder des registres à décalage pour en faire des circuits de transfert de données.
- Monter des bascules en circuits diviseurs de fréquence et en compteurs.
- Comprendre les caractéristiques types des bascules de Schmitt.
- Concevoir un oscillateur astable avec la minuterie 555.
- Reconnaître et prévoir les effets d'une dérive d'horloge dans les circuits synchrones.
- Dépanner divers types de circuits à bascules.

Introduction

Nous avons étudié jusqu'ici des circuits logiques combinatoires dont les sorties, à un instant donné, ne dépendent que des valeurs présentes sur les entrées. Toute condition antérieure

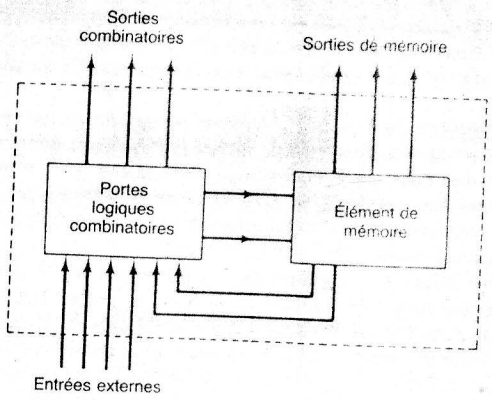


FIGURE 5-1 Schéma d'un système numérique général.

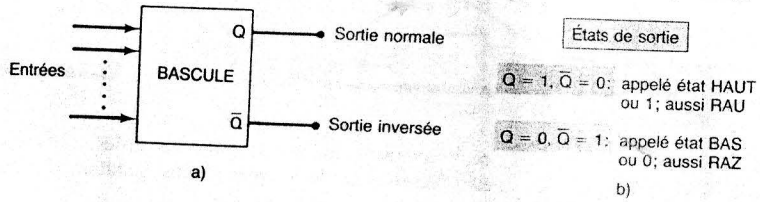
n'a aucun effet sur les valeurs actuelles des sorties, parce que les circuits combinatoires n'ont pas de mémoire. Dans la majorité des systèmes numériques, on retrouve une combinaison de circuits combinatoires et de dispositifs à mémoire.

La figure 5-1 illustre le schéma fonctionnel d'un système numérique général dans lequel on retrouve aussi bien des portes logiques que des dispositifs à mémoire. La section combinatoire est alimentée par des signaux d'entrée externes et par les sorties des dispositifs à mémoire. Le circuit combinatoire agit sur ces entrées pour produire diverses sorties, certaines servant à déterminer les valeurs binaires stockées dans les éléments de mémoire. La sortie de certains de ces éléments de mémoire revient comme entrée des circuits logiques de la partie combinatoire. Ceci est une indication que les sorties externes d'un système numérique dépendent autant des entrées externes que des informations mémorisées dans d'autres sections.

L'élément de mémorisation le plus important est la *bascule*, constituée d'un ensemble de portes logiques. Même si, en soi, une porte logique ne retient pas de donnée, il est possible d'en raccorder quelques-unes ensemble afin d'obtenir le stockage d'une information. Il existe différentes façons de monter les portes pour obtenir ces bascules.

La figure 5-2 a) est le symbole générique d'une bascule. On y voit deux sorties, désignées Q et $\overline{Q}$, qui sont l'inverse l'une de l'autre. En réalité, la lettre Q est arbitraire et on pourrait utiliser n'importe quelle autre lettre. La sortie Q a reçu le nom de sortie *normale* de la bascule, tandis que $\overline{Q}$ est appelée la sortie *inversée*. Chaque fois que l'on fait référence à l'état d'une bascule, on désigne la situation de sa sortie normale (Q); il est toujours entendu que la situation de la sortie inversée ($\overline{Q}$) est le complément de l'autre. Par exemple, si on dit que la bascule est dans l'état HAUT (1), on veut dire que sa sortie

FIGURE 5-2 Symbole général d'une bascule et définition des deux sorties possibles.



est $Q = 1$; si on dit qu'elle dans l'état BAS (0), cela signifie que $Q = 0$. Il va de soi que dans ces deux cas, l'état de $\overline{Q}$ est toujours l'inverse de Q.

Une bascule a donc, comme l'illustre la figure 5-2 b), deux états de fonctionnement. Observez les deux façons de désigner ces états. Apprenez-les bien car on y a recours abondamment dans les circuits numériques.

Le symbole de la figure 5-2 a) laisse entendre qu'une bascule peut avoir plusieurs entrées. Ces entrées sont employées pour commuter (« faire basculer ») la bascule entre ses deux états. Comme nous le verrons, une entrée de la bascule a simplement besoin de l'excitation d'une impulsion pour changer l'état de sortie de la bascule, et cette sortie conserve sa nouvelle valeur même après la disparition de l'impulsion. C'est cela qui fait de la bascule un dispositif de *mémorisation*.

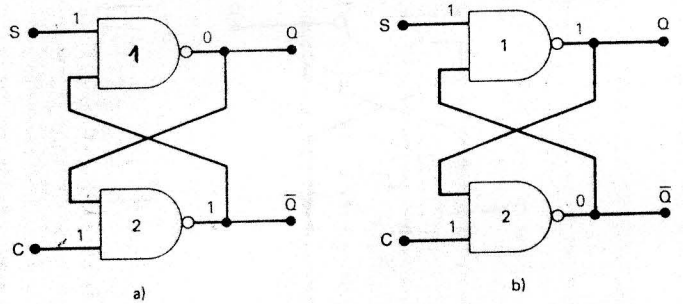
Le terme bascule a plusieurs synonymes, dont *élément de mémoire* et *multivibrateur bistable*. La première expression, « élément de mémoire », s'adresse à certaines bascules que nous allons décrire. La deuxième expression, « multivibrateur bistable », est le nom plus technique pour une bascule; on ne l'emploie presque pas à cause de sa longueur.

5-1 ÉLÉMENT DE MÉMOIRE EN PORTES NON-ET

La bascule la plus élémentaire est construite au moyen de deux portes NON-ET ou de deux portes NI. La version NON-ET, appelé *élément de mémoire en portes NON-ET* ou plus simplement *élément de mémoire*, est montrée à la figure 5-3 a). Les deux portes NON-ET sont rétrocouplées, de sorte que la sortie de la porte NON-ET 1 est connectée à l'une des entrées de la porte NON-ET 2, et vice versa. Les sorties, appelée Q et $\overline{Q}$ respectivement, sont les sorties de l'élément de mémoire. Dans des conditions normales, une sortie est toujours l'inverse de l'autre. Les entrées de l'élément de mémoire sont désignées S (d'après SET) et C (d'après CLEAR), pour des raisons qui deviendront évidentes plus loin.

Les entrées S et C se trouvent normalement toutes les deux au niveau HAUT, et l'on doit momentanément en porter une au niveau BAS pour changer l'état de sortie de la bascule. La première étape de notre analyse sera de montrer qu'il existe deux états également probables quand $S = 1$ et $C = 1$. Une des possibilités est illustrée à la figure 5-3 a) où $Q = 0$, $\overline{Q} = 1$. Quand $Q = 0$, les entrées de la porte NON-ET 2 sont 0 et 1, ce qui force $\overline{Q}$ à 1. Cette valeur 1 fait donc en sorte qu'il y a maintenant un 1 sur chacune des deux entrées de la porte NON-ET 1 produisant ainsi une sortie 0 (rappelons que la sortie est la borne Q). En effet, nous avons un niveau BAS à la sortie de la porte NON-ET 1, ce qui

FIGURE 5-3 Un élément de mémoire en NON-ET possède deux états stables possibles quand $S = C = 1$.



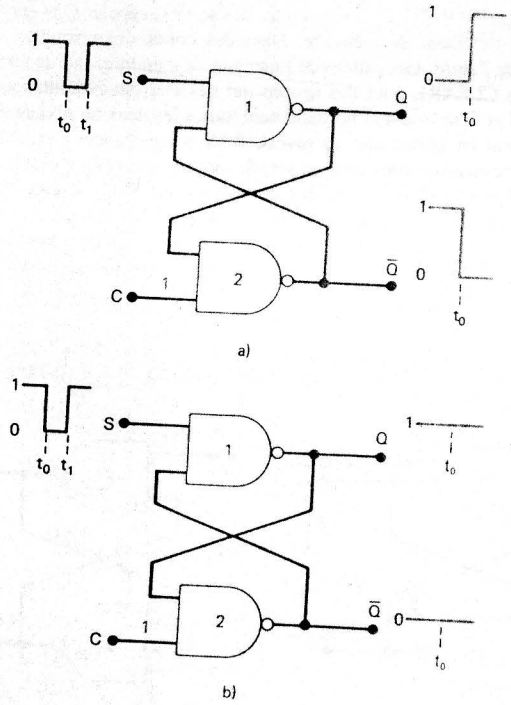
donne lieu à un niveau HAUT à la sortie de la porte NON-ET 2, sortie qui, en retour, garde la sortie de la porte NON-ET 1 à 0.

La seconde possibilité est celle de la figure 5-3 b) où $Q = 1$, $\overline{Q} = 0$. Le niveau HAUT à la sortie de la porte NON-ET 1 provoque un niveau BAS à la sortie de la porte NON-ET 2, sortie qui, en retour, garde la sortie de la NON-ET 1 au niveau HAUT. Quand $S = 1$ et $C = 1$, il y a donc deux possibilités; comme nous le verrons, l'état que prend effectivement la bascule dépend des valeurs appliquées précédemment aux entrées.

Mise à 1 de l'élément de mémoire Maintenant examinons ce qui se passe quand l'entrée S est momentanément amenée à 0 pendant que C reste à 1. La figure 5-4 a) montre ce qui se produit quand $Q = 0$ avant l'arrivée de l'impulsion. Au moment où S est amené au niveau BAS, à t_0, Q passe à 1, ce qui force $\overline{Q}$ à devenir 0: la porte NON-ET 1 a maintenant deux 0 à ses bornes d'entrée. On voit donc qu'à t_1, au moment où S revient au niveau HAUT, la sortie de la NON-ET 1 *reste* au niveau HAUT, ce qui a pour effet de garder la sortie de la NON-ET 2 au niveau BAS.

À la figure 5-4 b), on voit ce qui se passe quand $Q = 1$ et $\overline{Q} = 0$ avant l'application de l'impulsion à S. Étant donné que $\overline{Q} = 0$, cela garde la sortie de la NON-ET 1 au niveau HAUT, l'application d'une impulsion de niveau BAS sur S ne change rien. Ainsi, quand S revient à 1, la bascule se trouve-t-elle toujours dans l'état $Q = 1$ et $\overline{Q} = 0$.

FIGURE 5-4 L'application d'une impulsion 0 à l'entrée S met toujours la bascule dans l'état $Q = 1$; $\overline{Q} = 0$; a) la sortie est $Q = 0$ avant l'impulsion de mise à 1; b) $Q = 1$ avant cette impulsion.



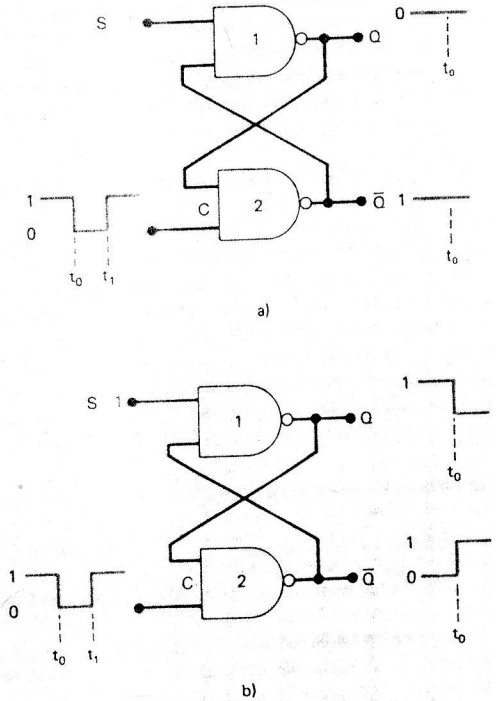
Il est possible de résumer la figure 5-4 en affirmant que l'application d'une impulsion de niveau BAS sur S place la bascule dans l'état $Q = 1$. On dit que cette opération *met* la bascule à 1. L'état $Q = 1$ est appelé l'**état 1** de la bascule (SET en anglais d'où le S)

Mise à zéro de l'élément de mémoire Voyons maintenant ce qui se produit quand l'entrée C est modifiée par une impulsion de niveau BAS pendant que S reste au niveau HAUT. La figure 5-5 a) nous fait voir ce qui se passe quand $Q = 0$ et $\overline{Q} = 1$ avant l'application de l'impulsion. Comme $Q = 0$, la sortie de la NON-ET 2 se trouve déjà au niveau HAUT, l'impulsion sur C n'a aucun effet. Quand C revient au niveau HAUT, la bascule à verrouillage est encore dans l'état $Q = 0$ et $\overline{Q} = 1$.

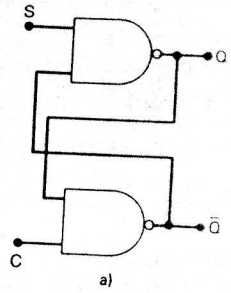
La figure 5-5 b) montre le cas lorsque $Q = 1$ à l'arrivée de l'impulsion sur C. Au moment où C reçoit une impulsion basse, à t_0, $\overline{Q}$ passe au niveau HAUT, ce qui force Q à 0 de sorte que la porte NON-ET 2 a un 0 appliqué sur chacune de ses entrées. Ainsi quand C revient à 1 après l'impulsion, à t_1, la sortie de la porte NON-ET 2 *reste* à 1, valeur qui, en retour, maintient la sortie de la porte NON-ET 1 au niveau BAS.

Nous pouvons résumer la figure 5-5 en énonçant le résultat suivant: l'application d'un niveau BAS à l'entrée C a toujours pour effet de mettre la bascule dans l'état $Q = 0$. On dit que cette opération *met à 0* la bascule; d'ailleurs l'état $Q = 0$ est appelé l'*état 0* (en anglais CLEAR d'où le C).

FIGURE 5-5 L'application d'une impulsion 0 à l'entrée C met toujours la bascule dans l'état $Q = 0$; $\overline{Q} = 1$; a) la sortie $Q = 0$ avant l'impulsion de mise à 0; b) la sortie $Q = 1$ avant cette impulsion.



Mise à 0 et à 1 simultanément Le dernier cas à considérer est celui où les entrées S et C sont mises toutes les deux à 0. Cette situation produit des niveaux HAUTS aux deux sorties de la bascule en NON-ET, de sorte que $Q = \overline{Q} = 1$. Évidemment, il s'agit là d'une condition indésirable, puisque les deux sorties doivent supposément toujours être l'inverse l'une de l'autre. De plus, quand les deux entrées S et C reviennent toutes les deux à 1, l'état de sortie de la bascule est conditionné par l'entrée qui revient à 1 en premier. Donc, des transitions simultanées à l'état 1 provoquent des résultats imprévisibles. Pour toutes ces raisons, la condition $C = S = 0$ est supposée ne jamais se produire dans un élément de mémoire en NON-ET.



a)

S	C	Sortie
1	1	inchangée
0	1	Q = 1
1	0	Q = 0
0	0	ambiguë

b)

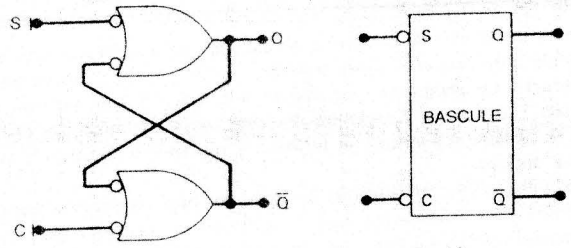
FIGURE 5-6 a) Élément de mémoire en NON-ET; b) sa table de vérité.

Récapitulation de l'élément de mémoire en NON-ET Toute la description précédente peut se mettre sous la forme d'une table de vérité (figure 5-6); voici un résumé de ce qui vient d'être dit:

1. $S = C = 1$; cette condition correspond à l'état normal de repos et elle n'affecte pas l'état de sortie de la bascule. Les sorties Q et $\overline{Q}$ demeurent dans l'état qu'elles occupaient avant l'application de cette condition d'entrée.
2. $S = 0, C = 1$; cette condition entraîne toujours la sortie dans l'état 1 où elle demeure même après le retour de S au niveau HAUT. On dit que c'est la condition de *mise à 1* de la mémoire.
3. $S = 1, C = 0$; cette condition entraîne toujours la sortie dans l'état 0 où elle demeure même après le retour de C au niveau HAUT. On dit que c'est la condition de *mise à 0* de la mémoire.
4. $S = C = 0$; cette condition est équivalente à vouloir mettre la mémoire à la fois à 1 et à 0, ce qui donne lieu à des résultats ambigus. Elle ne doit jamais servir.

Représentations synonymes D'après la description de la mémoire en NON-ET, on voit bien que les entrées S et C sont vraies au niveau BAS. L'entrée S quand elle passe à 0 met la bascule dans l'état $Q = 1$, alors que l'entrée C, quand elle passe à 0, met la bascule dans l'état $Q = 0$. C'est la raison pour laquelle le circuit de l'élément de mémoire en NON-ET est souvent tracé en recourant au symbole synonyme de la porte NON-ET, comme à la figure 5-7 a). Les ronds sur les lignes d'entrées S et C mettent en évidence le fait que ces entrées sont vraies au niveau BAS.

La figure 5-7 b) illustre un symbole rectangulaire que vous verrez souvent dans ce livre. Les ronds des entrées indiquent, comme toujours, qu'elles sont vraies au niveau



a) b)

FIGURE 5-7 a) Représentation équivalente d'une bascule en NON-ET; b) symbole rectangulaire plus simple.

BAS. Chaque fois que nous utiliserons ce rectangle, ce sera pour représenter un élément de mémoire en NON-ET.

EXEMPLE 5-1

Les formes d'ondes de la figure 5-8 sont appliquées aux entrées S et C d'une mémoire en NON-ET. Supposez qu'au départ $Q = 0$; trouvez la forme d'onde de Q.

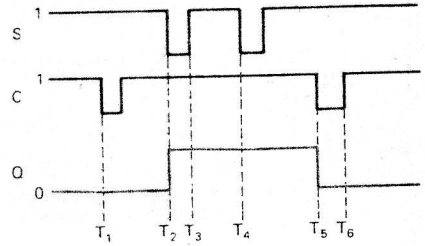


FIGURE 5-8 Exemple 5-1.

Solution Au départ $S = C = 1$, de sorte que Q demeure à 0. L'impulsion de niveau BAS qui est imposée à C à l'instant T_1 n'a aucun effet, puisque Q est déjà dans l'état 0.

La seule façon de faire passer Q à la valeur 1 est d'appliquer une impulsion basse sur l'entrée S. C'est la condition qui est réalisée à l'instant T_2 quand S passe pour la première fois au niveau BAS. Quand S revient à 1 à l'instant T_3, Q demeure au niveau HAUT.

À l'instant T_4, quand S revient une deuxième fois à 0, la bascule ne réagit pas puisqu'elle est déjà dans l'état 1.

La seule façon de ramener Q à l'état 0 est d'appliquer une impulsion basse sur l'entrée C. C'est d'ailleurs ce qui se produit en T_5. Quand C revient à 1, à l'instant T_6, Q reste dans son nouvel état BAS.

Cet exemple montre que la sortie de la mémoire « se souvient » de la dernière entrée validée et qu'elle ne passe à l'autre état que si l'entrée opposée est validée.

EXEMPLE 5-2

Il est pratiquement impossible de construire un interrupteur mécanique dans lequel il n'y aurait qu'une seule transition entre deux tensions, à cause du phénomène du **rebondissement des contacts**. Ce phénomène est illustré à la figure 5-9 a), où l'on voit que le passage du contact 1 au contact 2 engendre plusieurs tensions transitoires causées par le rebondissement de la lame (ouvertures et fermetures répétées du contact 2) avant la stabilisation de cette dernière.

Les multiples transitions qui affectent le signal de sortie durent généralement moins de quelques millisecondes, mais pour certaines applications c'est une durée inacceptable. Un élément de mémoire NON-ET peut être utilisé pour prévenir l'oscillation de la sortie. Décrivez le fonctionnement du circuit antirebond de la figure 5-9 b).

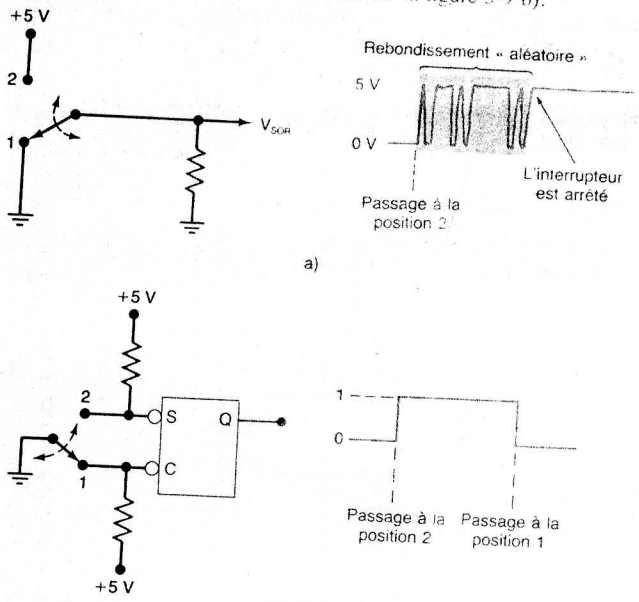


FIGURE 5-9 a) Le rebondissement des contacts mécaniques donne lieu à plusieurs transitions; b) un élément de mémoire en NON-ET peut éliminer ces rebonds.

Solution Supposons que l'interrupteur soit au début en position 1, de sorte que l'entrée C est au niveau BAS et $Q = 0$. Quand l'interrupteur est actionné et amené en position 2, C passe au niveau HAUT, et un niveau BAS apparaît sur l'entrée S au moment du contact initial de l'interrupteur. Cela a pour effet de placer, en l'espace de quelques nanosecondes (temps de réponse de la porte NON-ET), la sortie à $Q = 1$. Maintenant, si l'interrupteur rebondit les deux entrées S et C sont à un niveau HAUT, ce qui n'affecte en rien la valeur de Q, qui reste au niveau HAUT. On voit donc que Q reste inchangé malgré les rebonds de la lame sur le contact 2.

De même, quand l'interrupteur passe de la position 2 à la position 1, une tension BASSE se retrouve sur l'entrée C au moment du contact initial. Cela met à zéro Q; et cette sortie conserve cet état même si la lame rebondit avant de s'arrêter définitivement.

L'insertion de cette bascule fait en sorte que Q effectue une seule transition quand l'interrupteur change de position.

Terminologie Le fait de mettre à zéro une mémoire ($Q = 0$) s'appelle aussi *remise à zéro* de la bascule; on voit indifféremment ces deux termes en technologie numérique. En fait, il arrive souvent que la mémoire S-C soit appelée mémoire R-S, le R remplaçant le C.

QUESTIONS DE RÉVISION

1. Indiquez l'état normal de repos des entrées S et C. Donnez l'état vrai de chaque entrée.
2. Donnez les valeurs présentes sur Q et $\overline{Q}$ après la mise (remise) à zéro d'une bascule.
3. *Vrai ou faux*: On ne peut jamais utiliser l'entrée S pour avoir $Q = 0$.
4. Quand on met pour la première fois sous tension une bascule, il est impossible de savoir les états initiaux de Q et $\overline{Q}$. Que devons-nous faire pour nous assurer que l'état initial d'une mémoire en NON-ET est toujours $Q = 1$?

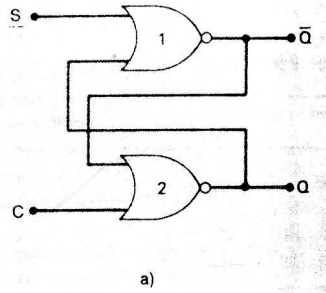
5-2 ÉLÉMENT DE MÉMOIRE EN PORTES NI

Deux portes NI rétrocouplées constituent une mémoire S-C. Un tel montage, illustré à la figure 5-10 a), est analogue à celui d'une mémoire en NON-ET, sauf que les sorties Q et $\overline{Q}$ sont maintenant interverties.

L'étude du fonctionnement d'une mémoire en NI se développe de manière tout à fait identique à celle de la mémoire en NON-ET. Les résultats sont donnés sous forme d'une table de vérité à la figure 5-10 b) et résumés ci-après:

1. $S = C = 0$: cette condition représente l'état normal de repos de la mémoire en NI et ne modifie en rien l'état de sortie. Q et $\overline{Q}$ demeurent dans l'état qu'elles occupaient avant l'arrivée de l'impulsion d'entrée.
2. $S = 1$, $C = 0$: cette condition a toujours pour effet de mettre Q à 1; état qui ne change pas même quand S revient à 0.

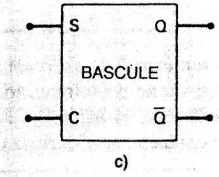
FIGURE 5-10 a) Bascule en NI; b) sa table de vérité; c) un symbole rectangulaire simplifié.



S	C	Sortie
0	0	inchangée
1	0	Q = 1
0	1	Q = 0
1	1	non valide*

* Produit $Q = \overline{Q} = 0$

b)



3. $S = 0, C = 1$; cette condition a toujours pour effet de mettre Q à 0; état qui ne change pas même quand C revient à 0.
4. $S = C = 1$; cette condition est équivalente à vouloir mettre la mémoire à 1 et à 0 en même temps et produit $Q = \overline{Q} = 0$. Si les deux entrées sont ramenées simultanément à 0, l'état de sortie résultante est imprévisible. Il ne faut jamais se servir de cette condition d'entrée.

L'élément de mémoire en NI fonctionne en tous points comme la mémoire en NON-ET, à l'exception des entrées S et C qui, maintenant, sont vraies au niveau HAUT plutôt qu'au niveau BAS, et de l'état normal de repos qui est $S = C = 0$. Q sera mis à 1 par une impulsion de niveau HAUT appliquée sur S et sera mis à 0 par une impulsion, toujours de niveau HAUT, sur C. La figure 5-10 c) nous fait voir le symbole rectangulaire de l'élément de mémoire en NI; dans ce symbole il n'y a pas de rond associé aux entrées. Ceci indique que les entrées sont vraies au niveau HAUT.

EXEMPLE 5-3

Supposez qu'au début $Q = 0$; déterminez la forme d'onde de Q d'une mémoire en NI quand les formes d'ondes appliquées à ses entrées sont celles de la figure 5-11.

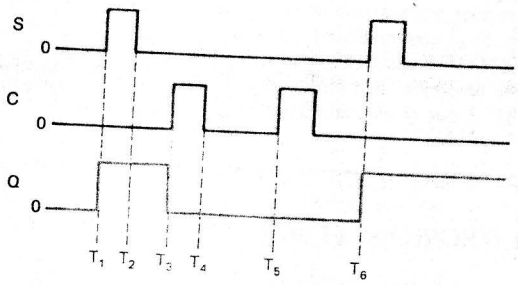


FIGURE 5-11 Exemple 5-3.

Solution Au début $S = C = 0$, ce qui n'a aucun effet sur Q qui reste à 0. Quand S passe au niveau HAUT, à l'instant T_1, Q est mis à 1 et reste à cette valeur même quand S revient à 0, à l'instant T_2.

À l'instant T_3, l'entrée C passe à 1 et met à zéro la sortie Q; la bascule reste à 0 même quand C revient à 0, à l'instant T_4.

Puisque la bascule est déjà dans l'état 0 à l'instant T_5, l'impulsion haute imposée à l'entrée C ne change rien. En T_6, l'impulsion sur S remet la mémoire à 1, et elle conserve cette valeur.

Cet exemple démontre que la bascule « se souvient » de la dernière entrée validée et qu'elle ne change d'état que lorsque l'entrée opposée est validée.

EXEMPLE 5-4

La figure 5.12 représente un circuit servant à détecter l'interruption d'un faisceau lumineux. Ce faisceau est dirigé sur un phototransistor monté en émetteur commun afin d'agir comme interrupteur. Supposez que la mémoire a été initialement mise à 0 par l'interrupteur S_1 qu'on a ouvert momentanément. Décrivez ce qui se produit si le faisceau est coupé momentanément.

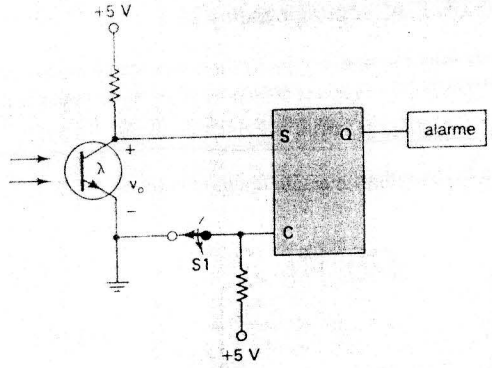


FIGURE 5-12 Exemple 5-4.

Solution Quand le transistor est illuminé par le faisceau, on suppose qu'il est saturé, de sorte que la résistance entre l'émetteur et le collecteur est très petite. C'est donc dire que v_0 est proche de 0 V. Il y a donc un niveau BAS sur la borne S et sur la borne C de la bascule.

Quand le faisceau est interrompu, le phototransistor ne conduit plus et sa résistance émetteur-collecteur devient très élevée (on a dans les faits un circuit ouvert). Dans un tel cas, v_0 est presque à 5 V, ce qui place un 1 sur la borne S; on sait qu'une telle valeur sur S met la mémoire à 1.

Cependant, la mémoire reste dans cet état même après la reprise du faisceau et le retour de v_0 à 0 V. L'alarme peut être arrêtée en plaçant un court instant un niveau HAUT sur l'entrée C.

Dans cette application, la faculté de mémorisation de l'élément est exploitée pour transformer un événement qui se produit un court instant (l'interruption du faisceau) en une condition constante.

État de la bascule à la mise sous tension Quand on applique la tension à un circuit, il est impossible de prédire l'état initial de la sortie de la bascule si les entrées S et C sont dans leur état indéterminé (par ex. $S = C = 1$ dans le cas de la mémoire NON-ET, ou $S = C = 0$ dans le cas de la mémoire NI). Il y a autant de chance que la sortie soit $Q = 1$ qu'elle soit $Q = 0$. Tout dépendra de facteurs comme les retards de propagation, la capacité parasite et la charge externe. Si un élément de mémoire ou une bascule doit se trouver dans un certain état au moment de la mise en marche d'un circuit afin d'en garantir le bon fonctionnement, il faut le mettre dans cet état en actionnant un court instant l'entrée S ou l'entrée C à la mise en marche du circuit. Ce qui est généralement réalisé par l'application d'une impulsion sur l'entrée appropriée.

QUESTIONS DE RÉVISION

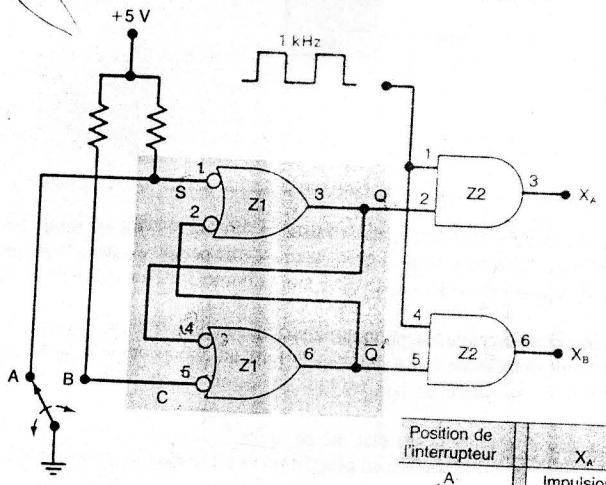
1. Nommez l'état normal de repos des entrées d'une mémoire en NI. Nommez l'état actif.
2. Quand une bascule est dans l'état 1, donnez les valeurs de Q et $\overline{Q}$.
3. Indiquez la seule manière de faire passer la sortie Q d'une mémoire en NI de 1 à 0.
4. Dites pourquoi le circuit de la figure 5-12 ne fonctionnerait par correctement si on subsituait à la mémoire en NI une mémoire en NON-ET.

5-3 ÉTUDE D'UN CAS DE DÉPANNAGE

Nous vous présentons deux exemples à titre d'illustration du genre de question à se poser quand on cherche à réparer un circuit contenant un élément de mémoire.

EXEMPLE 5-5

Analysez et décrivez le fonctionnement du circuit de la figure 5-13.



Position de l'interrupteur	X_A	X_B
A	Impulsions	BAS
B	BAS	Impulsions

FIGURE 5-13 Exemples 5-5 et 5-6.

Solution L'interrupteur sert à mettre à un ou à zéro la mémoire NON-ET en vue de produire des signaux francs (sans transitions dues aux rebonds) sur Q et $\overline{Q}$. Ces sorties déterminent à travers quelle porte ET passe le signal d'horloge de 1 kHz et sur quelle sortie, X_A ou X_B, on le retrouve.

Quand l'interrupteur est mis en position A, la mémoire est mise à un, $Q = 1$. Cela a pour effet de diriger le signal d'horloge de 1 kHz sur X_A, tandis que le niveau BAS de $\overline{Q}$ garde $X_B = 0$. Quand l'interrupteur passe sur la position B, la mémoire est mise à zéro, $Q = 0$, ce qui garde $X_A = 0$, tandis que le niveau HAUT de $\overline{Q}$ autorise le passage des impulsions jusqu'à X_B.

EXEMPLE 5-6

Un technicien a pour tâche de tester le circuit de la figure 5-13; il consigne les résultats suivants:

Position de l'interrupteur	S (Z1-1)	C (Z1-5)	Q (Z1-3)	$\overline{Q}$ (Z1-6)	X_A (Z2-3)	X_B (Z2-6)
A	BAS	HAUT	BAS	HAUT	BAS	Impulsions
B	HAUT	BAS	BAS	HAUT	BAS	Impulsions

Il remarque que lorsque l'interrupteur est en position B, le circuit fonctionne correctement, tandis que lorsqu'il est en position A, la mémoire n'active pas l'état $Q = 1$. Quelles sont les défaillances possibles pouvant expliquer ce mauvais fonctionnement?

Solution Il existe plusieurs possibilités:

1. Une connexion interne ouverte dans Z1-1. Cela aurait pour effet d'empêcher Q de réagir aux valeurs de l'entrée S.
2. Une panne d'un composant interne dans la porte en NON-ET Z1, ce qui l'empêcherait de réagir correctement.
3. La sortie Q reste en permanence au niveau BAS. Cela peut avoir comme cause:
   a) Z1-3 est court-circuité à la masse de l'intérieur.
   b) Z1-4 est court-circuité à la masse de l'intérieur.
   c) Z2-2 est court-circuité à la masse de l'intérieur.
   d) Le nœud Q est court-circuité à la masse de l'extérieur.

Une mesure avec un ohmmètre entre Q et la masse devrait déterminer si l'une quelconque de ces conditions est présente. Une observation devrait mettre en évidence tout court-circuit externe.

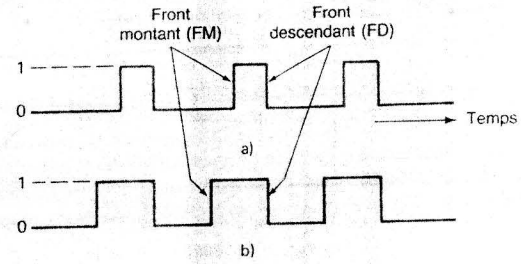
Pourquoi ne pas envisager un court-circuit de $\overline{Q}$ à V_{CC} par l'intérieur ou l'extérieur? Une simple réflexion nous amènera rapidement à la conclusion qu'il ne peut s'agir de ce genre de défaillance. Si $\overline{Q}$ était court-circuité à V_{CC}, cela n'empêcherait pas la sortie Q de passer à la valeur HAUTE quand S passe au niveau BAS. Comme Q ne prend *pas* la valeur HAUTE, cette possibilité doit être exclue. La raison pour laquelle $\overline{Q}$ semble figé sur la valeur HAUTE et que Q est, lui-même, figé sur la valeur BASSE et garde $\overline{Q}$ en permanence à 1.

5-4 SIGNAL D'HORLOGE ET BASCULES SYNCHRONES

Les circuits numériques peuvent fonctionner de façon soit **synchrone** ou **asynchrone**. Dans les systèmes asynchrones, la sortie des circuits logiques peut changer d'état à tout moment quand une ou plusieurs entrées changent. Un système asynchrone est difficile à concevoir et à dépanner.

Par contre dans un système synchrone, le moment exact où la sortie change d'état est commandé par un signal que l'on appelle couramment **signal d'horloge**. Ce signal est généralement un train d'ondes rectangulaires ou carrées, comme ceux de la figure 5-14. Le signal d'horloge est habituellement distribué à tous les étages du système, de sorte que la plupart des sorties du système, sinon toutes, changent d'état seulement quand le signal

FIGURE 5-14 Signaux d'horloge.



d'horloge effectue une transition. Ces transitions, appelées **fronts**, sont identifiées au moyen de flèches sur la figure 5-14. Quand le signal d'horloge passe de 0 à 1, on parle du **front montant** (transition positive); quand il passe de 1 à 0, on parle de **front descendant** (transition négative). Dans la suite de ce livre, nous utiliserons les abréviations FM et FD pour signifier front montant et front descendant, respectivement.

La majorité des systèmes numériques existants sont surtout des machines synchrones (même si certaines parties sont asynchrones), du fait que les circuits synchrones sont plus simples à concevoir et à dépanner. Leur dépannage est plus facile parce que leurs sorties ne peuvent changer qu'à des instants précis bien connus. Autrement dit, presque tous les changements sont synchronisés avec les transitions du signal d'horloge.

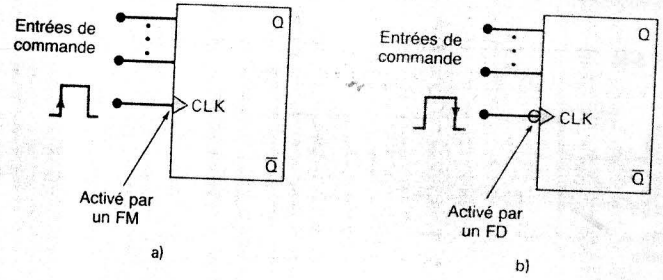
La synchronisation orchestrée par des signaux d'horloge est réalisée au moyen de *bascules synchrones* qui ont été étudiées pour changer d'état au moment de la transition associée à un front ou à l'autre du signal d'horloge.

Bascules synchrones Nous nous proposons d'étudier plusieurs types de bascules synchrones que l'on retrouve dans nombre d'applications. D'abord, passons en revue les principes communs à chacune d'elles.

1. Toutes les bascules synchrones disposent d'une entrée **d'horloge** que nous désignerons par l'abréviation *CLK* (voir figure 5-15). Cette entrée est symbolisée sur les schémas par un petit triangle qui signale que cette bascule ne réagit qu'au passage d'un des fronts du signal d'horloge. La bascule de la figure 5-15 a) est déclenchée exclusivement par un front montant (FM), tandis que celle de la figure 5-15 b) l'est exclusivement par un front descendant, comme l'indique d'ailleurs le petit rond devant le triangle de l'entrée *CLK*.
2. Toutes les bascules synchrones possèdent au moins une entrée de **commande**. Ces entrées sont désignées par différentes lettres, selon leur fonction. Ce sont ces entrées qui déterminent l'état de sortie de la bascule (comme les entrées *S* et *C*), mais cet état n'apparaît en sortie qu'au moment où se produit la transition dans le signal d'horloge. Autrement dit, leur effet est *synchronisé* avec l'application du signal d'horloge *CLK*. Pour cette raison, elles ont reçu le nom d'entrées **synchrones**.

   Par exemple, les entrées de commande de la bascule de la figure 5-15 a) n'ont *aucun* effet sur la sortie *Q* tant que le FM du signal d'horloge n'est pas arrivé. De même, les entrées de commande de la bascule de la figure 5-15 b) restent sans effet tant que le FD du signal d'horloge n'est pas arrivé.

FIGURE 5-15 Les bascules ont une entrée d'horloge: a) activée par le FM; b) activée par le FD. Les entrées de commande dictent la nature du changement.



3. Autrement dit, les niveaux logiques des entrées de commande dictent le changement qui doit paraître à la sortie de la bascule, tandis que le signal d'horloge déclenche ce changement au moment où arrive un des fronts.

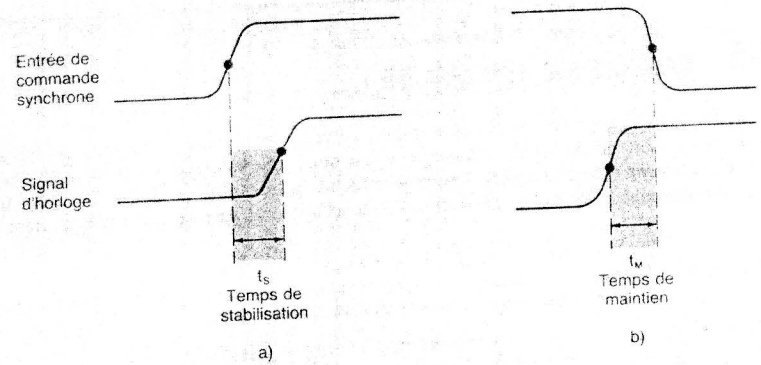


FIGURE 5-16 Les entrées de commande doivent rester stable pendant a) un temps t_S avant la transition active de l'horloge, et b) un temps t_M après la transition active de l'horloge.

Temps de stabilisation et temps de maintien Deux exigences de synchronisation doivent être respectées pour qu'une bascule synchrone réponde correctement à ses entrées de commande lorsqu'arrive le front déclencheur de *CLK*. Ces exigences sont représentées sous forme graphique sur la figure 5-16 pour une bascule déclenchée par un FM.

Le **temps de stabilisation, t_S,** est l'intervalle qui précède immédiatement le front déclencheur du signal d'horloge, pendant lequel l'entrée synchrone doit être gardée au niveau approprié. Les fabricants de CI spécifient généralement la durée de stabilisation minimale admissible. Si on ne respecte pas ce temps, il n'est pas garanti que la bascule répondra correctement à l'arrivée du front.

Le **temps de maintien, t_M,** est l'intervalle qui suit immédiatement le front déclencheur du signal d'horloge pendant lequel l'entrée synchrone doit être gardée au niveau approprié. Les fabricants de CI spécifient généralement le temps minimal acceptable. Si on ne respecte pas ce temps, la bascule ne sera pas déclenchée correctement.

D'après ce qui vient d'être dit, on voit que l'entrée de commande doit rester stable (inchangée) pendant une durée égale à la somme du temps t_S, qui *précède* le front déclencheur, et du temps t_M, qui *suit* ce même front.

Les bascules de CI ont des temps t_S et t_M de l'ordre des nanosecondes. Ainsi le temps de stabilisation est souvent compris entre 5 et 50 ns, alors que le temps de maintien va de 0 à 10 ns. Notez que ces temps sont mesurés à 50 pour cent de la hauteur (mi-hauteur) des fronts.

Comme nous le verrons, il est très important de respecter ces conditions de synchronisation dans les systèmes synchrones, parce qu'il arrive souvent que les entrées synchrones changent presque au même instant que l'entrée *CLK*.

QUESTIONS DE RÉVISION

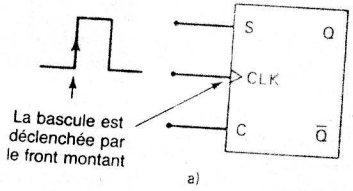
1. Indiquez les deux types d'entrées dans une bascule synchrone.
2. Que signifie la mention « déclenchée par un front ».

3. *Vrai ou faux:* L'entrée *CLK* agit sur la sortie d'une bascule seulement quand survient une transition active dans l'entrée de commande.
4. Définissez temps de stabilisation et temps de maintien d'une bascule synchrone.

5-5 BASCULE S-C SYNCHRONE

La figure 5-17 a) reproduit le symbole logique d'une *bascule S-C synchrone* déclenchée par le front montant du signal d'horloge. C'est donc une bascule qui passe d'un état à l'autre *seulement* quand le signal d'horloge effectue la transition de 0 à 1. Les entrées S et C commandent la valeur qu'on retrouve à la sortie de la bascule de la même manière

FIGURE 5-17 a) Bascule S-C synchrone qui réagit au front montant de l'impulsion d'horloge; b) sa table de vérité; c) formes d'onde caractéristiques.

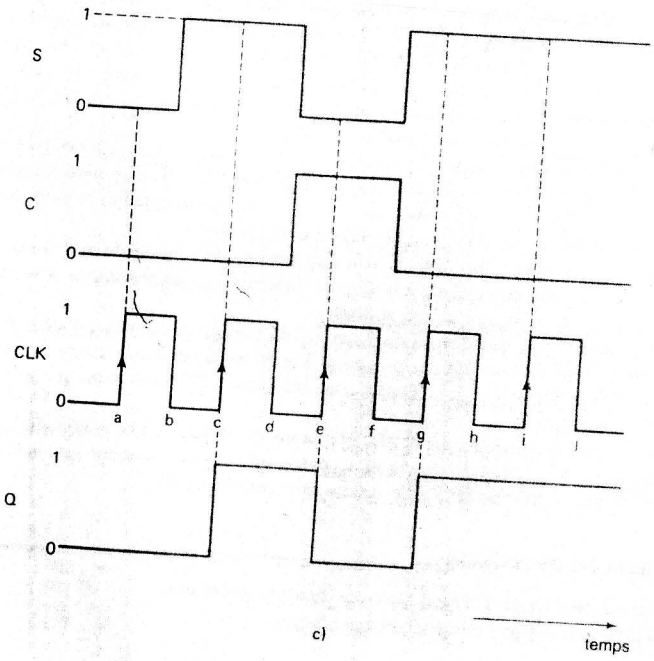


a)

S	C	CLK	Q
0	0	↑	Q_0 (inchangée)
1	0	↑	1
0	1	↑	0
1	1	↑	Ambiguë

Q_0 le niveau de sortie avant le FM de l'horloge

b)



c)

que dans le cas d'un élément de mémoire NI, sauf qu'ici, la bascule ne modifie pas sa sortie avant qu'arrive le front montant du signal d'horloge.

La table de vérité de la figure 5-17 b) montre comment la sortie d'une bascule réagit au front montant du signal d'horloge pour les diverses combinaisons des entrées S et C. La flèche vers le haut (↑) indique l'instant du front montant du signal d'horloge *CLK*; le symbole Q_0 signifie le niveau de Q avant le passage du front montant. Les fabricants de CI ont recours intensivement à cette convention dans leurs fiches techniques.

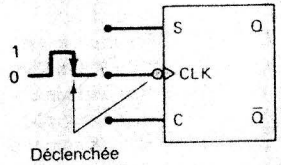
Les formes d'ondes que nous pouvons observer sur la figure 5-17 c) illustrent le fonctionnement de la bascule S-C synchrone. Si nous présumons que les temps de stabilisation et de maintien sont respectés dans tous les cas, l'analyse des formes d'ondes est la suivante:

1. Au départ, toutes les entrées sont à 0 et on suppose que la sortie Q est aussi à 0.
2. Quand le FM de la première impulsion d'horloge arrive (point *a*), les entrées S et C sont toutes les deux à 0 de sorte que la sortie de la bascule demeure dans l'état $Q = 0$ ($Q = Q_0$).
3. À l'arrivée du FM de la deuxième impulsion d'horloge (point *c*), l'entrée S a maintenant la valeur 1 tandis que C est restée à 0. La bascule se met à 1 quand arrive le front montant de l'impulsion d'horloge.
4. Quand la troisième impulsion présente son front montant (point *e*), il se trouve que $S = 0$ et $C = 1$, ce qui met à 0 la bascule.
5. La quatrième impulsion met à 1 la bascule (point *g*), car $S = 1$ et $C = 0$ quand survient le front montant.
6. Le front montant de la cinquième impulsion arrive quand S et C ont les mêmes valeurs qu'en 5 ce qui ne modifie en rien la sortie qui reste à 1.
7. La condition $S = C = 1$ ne doit pas être utilisée parce qu'elle donne des résultats ambigus.

Il est bon de noter sur cette figure que les fronts descendants n'affectent pas la sortie de la bascule. De plus, vous devez avoir remarqué que les entrées S et C ne modifient en rien la sortie de la bascule tant que le front montant du signal d'horloge n'est pas arrivé. S et C sont les entrées de *commande* synchrones: ce sont elles qui dictent à la bascule l'état qu'elle devra occuper lorsque surviendra le front montant du signal d'horloge. L'entrée *CLK* est celle qui déclenche le changement dans la bascule, dicté par les valeurs appliquées sur S et C.

La figure 5-18 nous fait voir le symbole d'une bascule S-C déclenchée par le front descendant du signal d'horloge et sa table de vérité. Le petit rond devant le triangle signale que la bascule réagit aux commandes seulement quand l'entrée *CLK* effectue la transition de 1 à 0. Cette bascule fonctionne en tous points comme la bascule à front montant, sauf

FIGURE 5-18 Bascule S-C synchrone déclenchée par le front descendant.



S	C	CLK	Q
0	0	↓	Q_0 (inchangée)
1	0	↓	1
0	1	↓	0
1	1	↓	Ambiguë

qu'ici la sortie ne change qu'au moment des fronts descendants (points *b*, *d*, *f*, *h* et *j* sur la figure 5-17). Dans les systèmes numériques, on retrouve aussi bien des bascules à front montant que des bascules à front descendant.

Circuits internes d'une bascule S-C déclenchée par un signal d'horloge Nous ne tenons pas à entrer dans le détail des circuits internes d'une bascule synchrone, parce que toutes les bascules sont intégrées dans des puces et que ce qui nous intéresse surtout c'est leur fonctionnement externe. Toutefois, nous pensons que l'examen d'une version simplifiée des circuits internes d'une bascule S-C peut être instructif. La figure 5-19 nous montre une bascule S-C déclenchée par un front.

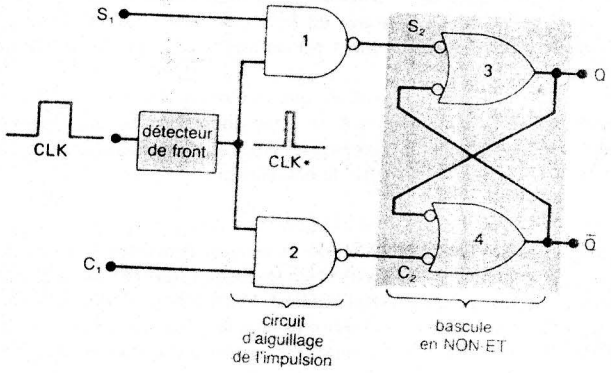


FIGURE 5-19 Version simplifiée des circuits internes d'une bascule S-C déclenchée par un signal d'horloge.

On retrouve dans un tel circuit les trois sections que voici:

1. Une bascule en NON-ET formée par les portes NON-ET 3 et 4.
2. Un circuit d'**aiguillage de l'impulsion** constitué par les portes NON-ET 1 et 2.
3. Un circuit **détecteur de front**.

Le circuit détecteur de front génère une impulsion montante étroite (*CLK**) qui se produit en coïncidence avec le front actif de l'impulsion d'horloge *CLK*. Le circuit d'aiguillage de l'impulsion « dirige » la pointe de tension sur l'entrée S_2 ou l'entrée C_2 selon les valeurs présentes sur les bornes S_1 et C_1. Par exemple, quand $S_1 = 1$ et $C_1 = 0$, le signal *CLK** est inversé et franchit la porte NON-ET 1 pour produire une impulsion BASSE sur l'entrée S_2 de façon à mettre l'élément de mémoire à $Q = 1$. Quand $S_1 = 0$ et $C_1 = 1$, le signal *CLK** est aussi inversé et franchit la porte NON-ET 2 pour venir appliquer une impulsion BASSE sur l'entrée C_2 et mettre l'élément de mémoire à $Q = 0$.

La figure 5-20 a) montre comment le signal *CLK** est produit dans le cas des bascules déclenchées par le front montant d'un signal d'horloge. L'INVERSEUR introduit un retard de quelques nanosecondes de sorte que les fronts de $\overline{CLK}$ ont lieu légèrement plus tard que ceux de *CLK*. La porte ET génère une pointe de tension de sortie qui est à 1 pendant seulement quelques nanosecondes, soit pendant le temps où $\overline{CLK}$ et *CLK* sont tous les deux au niveau HAUT. Le résultat est une impulsion étroite sur *CLK** qui apparaît au moment du front montant de *CLK*. Le montage de la figure 5-20 b), de la même façon, produit

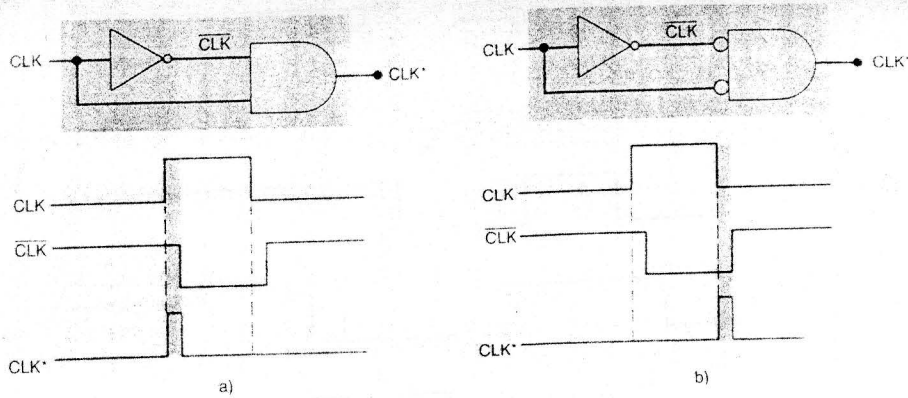


FIGURE 5-20 Circuits détecteurs de fronts construits pour des bascules déclenchées par a) le front montant du signal; b) le front descendant du signal.

*CLK** au moment du front descendant de l'horloge; on l'utilise pour le déclenchement des bascules commandées par un front descendant.

Comme le signal *CLK** n'est au niveau HAUT que quelques nanosecondes, *Q* n'est affecté par les niveaux sur S_1 et C_1 qu'un court instant durant et après le passage du front actif de *CLK*. C'est ce phénomène qui dote la bascule de cette faculté d'être déclenchée par un signal.

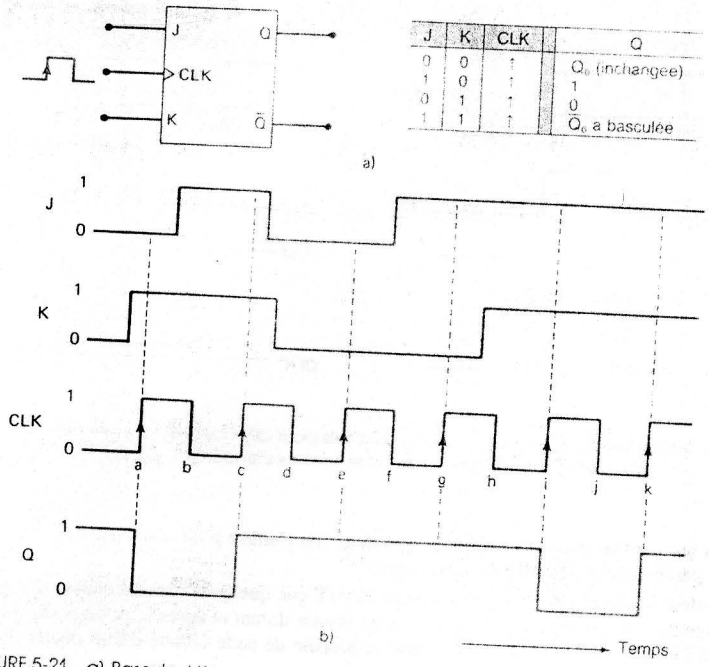
QUESTIONS DE RÉVISION

1. Supposons que les formes d'ondes de la figure 5-17 c) sont appliquées aux entrées de la bascule de la figure 5-18. Déterminez ce qui arrive au point *Q* en *b*.
2. Expliquez comment les entrées *S* et *C* influencent *Q* durant le front actif de *CLK*.

5-6 BASCULE J-K SYNCHRONE

On peut voir à la figure 5-21 a) une *bascule J-K synchrone* déclenchée par le front montant d'un signal d'horloge. Les entrées *J* et *K* commandent l'état de la bascule comme le font les entrées *S* et *C* de la bascule S-C synchrone, à l'exception de l'importante différence que voici: *la condition J = K = 1 ne donne pas lieu à une condition ambiguë*. Quand survient cette condition, la bascule passe toujours à l'état opposé à l'arrivée du front montant du signal d'horloge. On dit que c'est le mode de **basculement**. Dans ce mode, si on laisse les entrées *J* et *K* toutes les deux au niveau HAUT, la bascule va passer à l'état opposé à chaque signal d'horloge.

La table de vérité reproduite à la figure 5-21 a) récapitule la réponse d'une bascule J-K quand arrive un FM, pour toutes les combinaisons de valeurs de *J* et *K*. Notez que cette table de vérité est identique à celle de la bascule S-C (figure 5-17), *sauf* pour la réponse à la condition $J = K = 1$. Cette condition a pour résultat $Q_0 = \overline{Q}_0$, ce qui signifie que la nouvelle valeur de *Q* est l'inverse de celle existant avant le front montant. Il y a eu basculement.



J	K	CLK	Q
0	0	↑	Q_0 (inchangée)
1	0	↑	1
0	1	↑	0
1	1	↑	$\overline{Q}_0$ a basculée

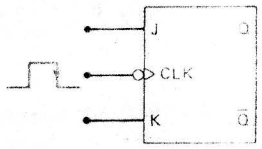
FIGURE 5-21 a) Bascule J-K synchrone réagissant à l'arrivée du front montant; b) formes d'ondes.

Les formes d'ondes de la figure 5-21 b) illustrent le fonctionnement de cette bascule. Une nouvelle fois, nous présumons que les temps de stabilisation et de maintien sont respectés.

1. Au départ, toutes les entrées sont à 0 et, par hypothèse, la sortie Q est à 1.
2. Quand arrive le front montant de la première impulsion d'horloge (point *a*) $J = 0$ et $K = 1$. Dans cette condition, la bascule est mise à 0.
3. Quand arrive la deuxième impulsion $J = 1$ et $K = 1$ (point *c*), de sorte qu'au moment de la transition montante, la bascule passe à l'état opposé, soit $Q = 1$.
4. Au point *e* de la forme de l'onde d'horloge, J et K sont tous les deux à 0 et la bascule reste dans le même état pendant cette transition.
5. Au point *g*, $J = 1$ et $K = 0$. Nous reconnaissons là la condition qui met à 1 la bascule. Toutefois, comme elle est déjà à 1, son état demeure inchangé.
6. Au point *i*, $J = K = 1$, ce qui fait passer la bascule dans son état opposé. La même chose se produit au point *k*.

Il convient de remarquer à la lumière de cet exemple que cette bascule ne réagit aucunement aux fronts descendants du signal d'horloge. De plus, on voit que les entrées J et K ne l'affectent pas sauf au moment du front montant. Utilisées seules, les entrées J et K ne peuvent pas modifier l'état de la bascule.

Le symbole de la bascule J-K synchrone déclenchée par le front descendant est montré à la figure 5-22. Le petit rond à l'entrée *CLK* indique une bascule qui est déclenchée quand



J	K	CLK	Q
0	0	↓	Q_0 (inchangée)
1	0	↓	1
0	1	↓	0
1	1	↓	$\overline{Q}_0$ a basculée

FIGURE 5-22 Bascule J-K déclenchée par le front descendant.

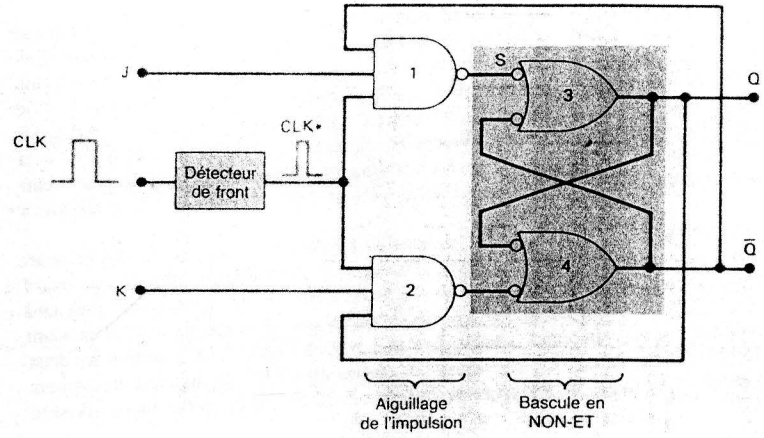
a lieu la transition de 1 à 0 du signal d'horloge. Cette bascule affiche un fonctionnement tout à fait identique à celui de la bascule à front montant de la figure 5-21, sauf que les changements d'états se font au passage du front descendant (points *b, d, f, h et j*). Ces deux types de bascules J-K déclenchées par un signal d'horloge sont courantes dans les systèmes.

La bascule J-K est plus polyvalente que la bascule S-C du fait qu'elle ne possède pas d'état ambigu. La condition $J = K = 1$ qui donne lieu au basculement de l'état de sortie est exploitée intensivement dans tous les types de compteurs binaires. Dans les faits, la J-K fait tout ce que fait la S-C, *plus* le basculement.

Circuits internes d'une bascule J-K déclenchée par un signal d'horloge On peut voir à la figure 5-23 une version simplifiée des circuits internes d'une bascule J-K déclenchée par un signal d'horloge. Nous voyons très bien les mêmes trois sections que dans la bascule S-C déclenchée par un signal d'horloge (figure 5-19). En fait, la seule différence entre ces deux circuits vient des sorties Q et $\overline{Q}$ qui sont ramenées sur des entrées des portes NON-ET d'aiguillage de l'impulsion. Cette connexion en rétroaction dote la bascule J-K du mode de basculement au moment de la condition $J = K = 1$.

Examinons plus en détail ce mode de basculement: supposons qu'à l'arrivée de l'impulsion *CLK*, $J = K = 1$ et $Q = 0$. Avec $Q = 0$, $\overline{Q} = 1$, la porte NON-ET 1 dirige *CLK** (inversée) sur l'entrée S de la bascule en NON-ET pour donner $Q = 1$. Si nous présumons que Q est au niveau HAUT quand arrive une impulsion d'horloge, la porte NON-ET 2 dirige *CLK** (inversée) sur l'entrée C de la mémoire pour donner $Q = 0$. Ainsi, Q se retrouve toujours dans l'état opposé à l'état précédent.

FIGURE 5-23 Circuits internes d'une bascule J-K déclenchée par un signal d'horloge



Pour que le basculement se produise comme nous venons de le décrire, il faut que l'impulsion *CLK** soit très étroite. Elle doit retourner à 0 avant que les sorties Q et $\overline{Q}$ basculent; sans cela, l'impulsion *CLK** et les nouvelles valeurs Q et $\overline{Q}$ feront basculer de nouveau le circuit.

QUESTIONS DE RÉVISION

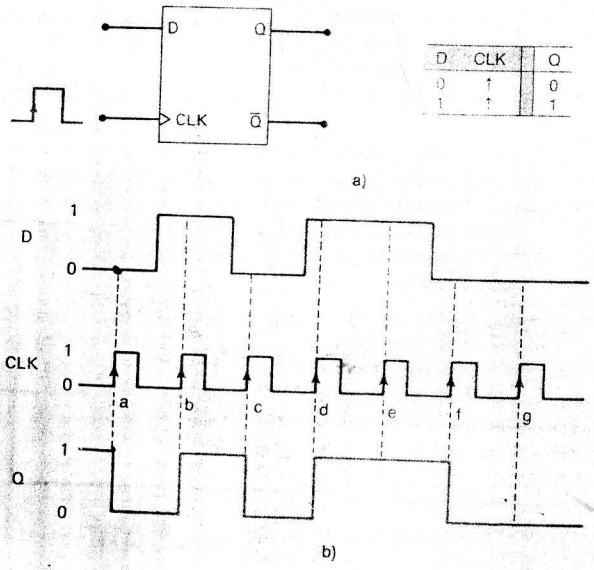
1. *Vrai ou faux:* Une bascule J-K peut remplacer une bascule S-C, mais une S-C ne peut être substituée à une J-K.
2. Une bascule J-K peut-elle avoir des conditions d'entrée ambiguës?
3. Donnez la condition J-K qui met Q à 1 quand s'effectue la transition d'horloge.

5-7 BASCULE D SYNCHRONE

La figure 5-24 a) nous fait voir le symbole d'une bascule D synchrone déclenchée par le front montant du signal d'horloge. Contrairement aux bascules S-C et J-K, cette bascule ne possède qu'une entrée de commande synchrone, appelée D, pour *données*. Le fonctionnement de la bascule D est très simple: Q prend l'état de l'entrée D à l'instant du front montant de *CLK*. Autrement dit, le niveau actuellement sur D se retrouvera *mémorisé* dans la bascule à l'instant du front montant. Les formes d'ondes de la figure 5-24 b) illustrent cette situation.

Supposons que la valeur initiale de Q soit le niveau HAUT. À l'instant du premier front montant, au point a, l'entrée D est BASSE; Q passera donc à l'état 0. Même si le niveau de l'entrée D change entre a et b, la sortie Q n'en est aucunement affectée, et Q continue de garder le niveau BAS que possédait l'entrée D au moment a. Au moment du

FIGURE 5-24 a) Bascule D déclenchée par un signal d'horloge; b) formes d'ondes.



D	CLK	Q
0	↑	0
1	↑	1

front montant en b, Q passe au niveau HAUT, puisque c'est le niveau de D à cet instant. Q mémorise la valeur HAUTE jusqu'au prochain front montant, puis passe au niveau BAS, à l'instant c, étant donnée que c'est la valeur de D à ce moment-là. De la même façon, la sortie Q épouse les niveaux présents sur D à l'instant des fronts montants correspondant aux points d, e et f. Notez qu'au point e la sortie Q reste au niveau HAUT, puisque c'est le niveau de D à cet instant.

Encore une fois, il est important de rappeler que Q ne change de valeur que si un front montant survient. L'entrée D n'a aucun effet entre deux fronts montants.

Une bascule D déclenchée par un front descendant fonctionne exactement comme on vient de le décrire, à la seule exception que le passage de la valeur de D sur Q survient aux moments des fronts descendants de *CLK*. Le symbole d'un bascule déclenchée par un front descendant contient un petit rond sur l'entrée d'horloge *CLK*.

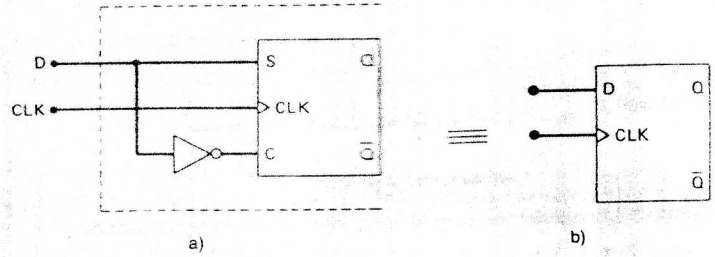


FIGURE 5-25 Bascule D déclenchée par un signal d'horloge réalisée à partir d'une bascule S-C.

Réalisation d'une bascule D Une bascule D déclenchée par un signal d'horloge n'est pas difficile à construire: il suffit d'ajouter un INVERSEUR à une bascule S-C déclenchée par un signal d'horloge, comme à la figure 5-25. Si vous testez les deux valeurs de D, vous constaterez que Q s'aligne sur la valeur en D à l'arrivée du front montant.

EXEMPLE 5-7

Expliquez comment on peut modifier une bascule J-K pour obtenir une bascule D.

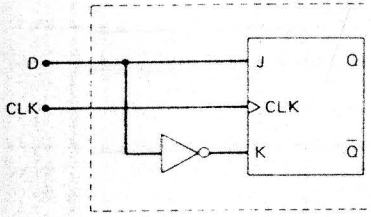
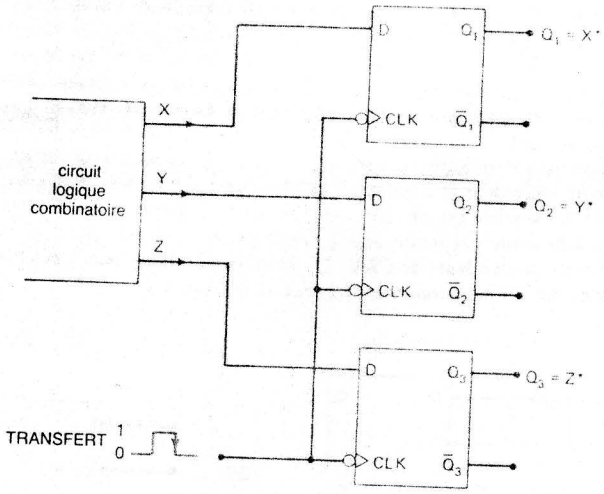


FIGURE 5-26 Bascule D déclenchée par un signal d'horloge réalisée à partir d'une bascule J-K.

Solution Cette modification est analogue à celle apportée à la bascule S-C de la figure 5-25. Voyez la solution à la figure 5-26.

Transfert de données en parallèle Vous vous demandez sans doute à quoi peuvent bien servir les bascules D, puisqu'il semble bien que leur sortie Q soit identique à l'en-



*après le passage d'un FD

FIGURE 5-27 Transfert en parallèle de données binaires à des bascules D.

trée D. Identique, pas vraiment, étant donné que Q n'est égal à D qu'à certaines occasions. Par exemple, voir les formes d'ondes de la figure 5-24.

Dans la plupart des applications des bascules D, la sortie Q prend la valeur de son entrée D à des moments bien précis. On peut voir un exemple de cela à la figure 5-27. Les sorties logiques X, Y et Z doivent être passées aux bornes Q_1, Q_2 et Q_3 en vue de leur stockage. En utilisant des bascules D, les niveaux appliqués à X, Y et Z sont reportés respectivement sur Q_1, Q_2 et Q_3 quand une impulsion de TRANSFERT est appliquée aux entrées *CLK*. Les bascules mémorisent ces valeurs pour un traitement ultérieur. Nous vous avons montré là un exemple de transfert *en parallèle* de données binaires, car X, Y et Z ont été passés ***simultanément*** aux bascules.

QUESTIONS DE RÉVISION

1. Dites ce qui arrive à la forme d'onde Q de la figure 5-24 b) si l'entrée D est toujours gardée à 0.
2. *Vrai ou faux:* La sortie Q est égale à l'entrée D en tout temps.

5-8 ÉLÉMENT DE MÉMOIRE D

La bascule D déclenchée par un signal d'horloge possède un circuit détecteur de front qui garantit que la sortie prend la valeur de l'entrée D seulement quand se produit la transition de déclenchement du signal d'horloge. Si ce détecteur est enlevé, on obtient un circuit qui fonctionne assez différemment. Ce circuit est appelé *un élément de mémoire D*; voyez son schéma à la figure 5-28 a).

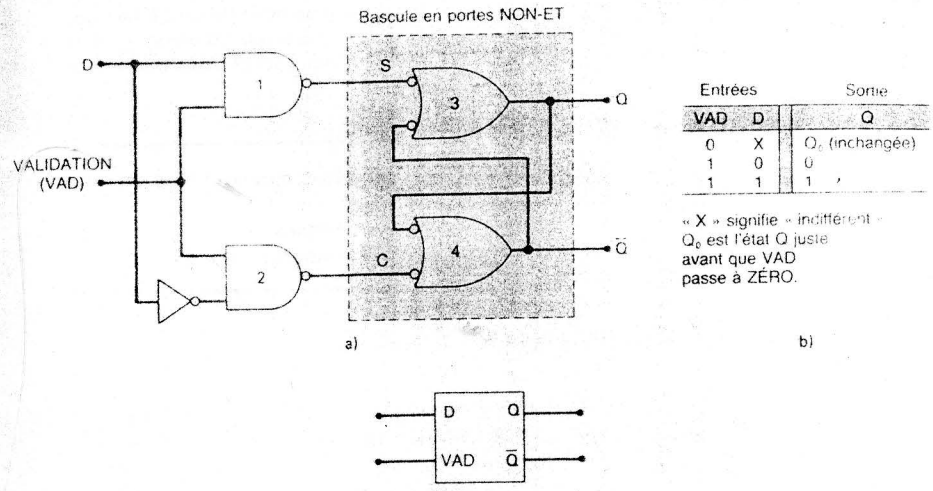


FIGURE 5-28 Élément de mémoire D: a) structure; b) table de vérité; c) symbole logique.

Ce circuit est constitué d'un élément de mémoire NON-ET et de deux portes NON-ET 1 et 2 pour l'aiguillage. L'entrée commune aux portes d'aiguillage a reçu le nom d'entrée de *validation* (abréviation *VAD*), et elle se distingue d'une entrée d'horloge puisque son effet sur les sorties Q et $\overline{Q}$ n'agit pas seulement à l'instant du passage de ses fronts. Le fonctionnement de l'élément de mémoire D est décrit ci-après:

1. Quand *VAD* est au niveau HAUT, l'entrée D produit un niveau BAS sur l'entrée S ou l'entrée C de l'élément de mémoire NON-ET afin de reporter sur Q le même niveau se trouvant sur D. Si D change pendant que *VAD* est HAUT, Q suit fidèlement les changements. Autrement dit, quand $VAD = 1$, la sortie Q a la même forme d'onde que D; dans ce mode, on dit que l'élément de mémoire D est « transparent ».
2. Quand *VAD* passe au niveau BAS, l'entrée D est empêchée d'influer sur l'élément de mémoire NON-ET, étant donné que les sorties des deux portes d'aiguillage sont forcées à la valeur 1. Pour cela, Q et $\overline{Q}$ demeurent au niveau qu'ils avaient jusqu'avant que *VAD* passe au niveau BAS. En d'autres mots, les sorties sont « verrouillées » sur leur niveau actuel et ne peuvent changer tant que *VAD* reste au niveau BAS, même si D change.

Ce fonctionnement est résumé dans la table de vérité de la figure 5-28 b). Le symbole logique d'un élément de mémoire D est représenté sur la figure 5-28 c). Observez que même si l'action de *VAD* ressemble beaucoup à celle de l'horloge d'une bascule déclenchée par un signal, il n'y a pas de petit triangle à l'entrée *VAD*. La raison en est que le petit triangle est réservée exclusivement aux entrées qui initient un changement à l'instant du passage d'un de leurs fronts.

EXEMPLE 5-8

Déterminez la forme d'onde de Q d'un élément de mémoire D pour les entrées *VAD* et D illustrées à la figure 5-29. Supposez que $Q = 0$ initialement.

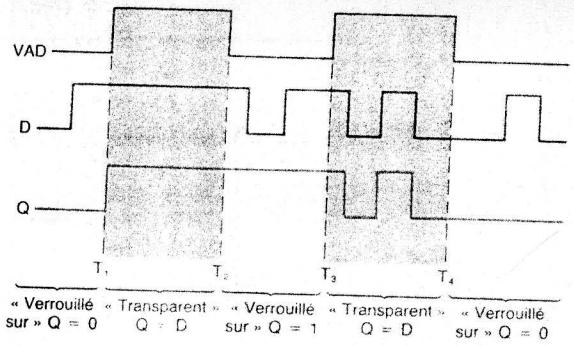


FIGURE 5-29 Formes d'ondes de l'exemple 5-8; voyez les deux modes de l'élément de mémoire *D*.

Solution Avant T_1, *VAD* est au niveau BAS, de sorte que Q est « verrouillé » sur le niveau courant 0 et reste fixe malgré les changements du niveau de D. Durant l'intervalle T_1 à T_2, *VAD* est au niveau HAUT, de sorte que Q varie exactement comme le signal sur D. Donc, Q passe au niveau HAUT en T_1 et y demeure puisque D ne change pas. Quand *VAD* revient à 0, en T_2, Q est verrouillé sur le niveau HAUT, celui qu'il avait en T_2 et reste à ce niveau tant que *VAD* est au niveau BAS.

En T_3, quand *VAD* repasse au niveau *HAUT*, Q reflète les changements qui interviennent sur D, et cela jusqu'au retour de *VAD* au niveau BAS. Pendant l'intervalle T_3 à T_4, la mémoire D est « transparente », étant donné que les variations se produisant sur D sont transmises intégralement à la sortie Q. En T_4, à l'instant où *VAD* passe au niveau BAS, Q se verrouille sur le niveau 0, le niveau de la sortie à l'instant T_4. Après T_4, les variations du signal sur D ne sont plus reflétées sur Q, puisque la sortie est verrouillée par le fait que $VAD = 0$.

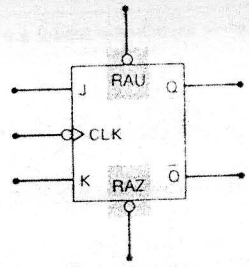
QUESTIONS DE RÉVISION

1. Indiquez la différence de fonctionnement entre un élément de mémoire *D* et une bascule *D* déclenchée par un signal
2. *Vrai ou faux:* Un élément de mémoire *D* est dans son mode transparent quand $VAD = 0$.
3. *Vrai ou faux:* Dans un élément de mémoire *D*, l'entrée *D* agit sur *Q* seulement quand $VAD = 1$.

5-9 ENTRÉES ASYNCHRONES

Dans les bascules synchrones précédentes, nous avons parlé d'entrées de *commande* pour les entrées *S*, *C*, *J*, *K* et *D*. Ces entrées sont également qualifiées de **synchrones** étant donné que la sortie de la bascule est synchronisée par le signal d'horloge. Comme nous l'avons vu, les entrées de commande synchrones sont utilisées concurremment à un signal d'horloge qui déclenche la réponse de la bascule.

La majeure partie des bascules synchrones possèdent en plus des **entrées asynchrones** qui agissent indépendamment des entrées synchrones et du signal d'horloge. On a recours à de telles entrées pour forcer en tout temps la remise à 1 ou à 0 de la bascule, quelles que soient les conditions des entrées. Une autre façon de présenter ces entrées est de dire



RAU	RAZ	Réponse de la bascule
1	1	Fonctionnement synchrone*
0	1	Q = 1
1	0	Q = 0
0	0	Inutilisée

* Q est fixé par J, K et CLK

FIGURE 5-30 Bascule J-K synchrone munie d'entrées asynchrones.

que ce sont des entrées **prioritaires**, qui imposent un état à la bascule malgré les commandes lancées par les autres entrées.

La figure 5-30 illustre une bascule J-K munie de deux entrées asynchrones désignées RAU (REMISE À UN) et RAZ (REMISE À ZÉRO). Ce sont des entrées vraies au niveau BAS, comme on peut le déduire par les petits ronds associés à ces entrées. La table à droite de ce symbole résume la réaction de la sortie de la bascule. Examinons ces différents cas:

- $RAU = RAZ = 1$: Les deux entrées asynchrones sont inactives et la bascule est libre de répondre aux commandes des entrées *J*, *K* et *CLK*; autrement dit, la réponse est la même que celle d'une bascule synchrone.
- $RAU = 0; RAZ = 1$: La borne RAU est vraie et Q est *immédiatement* mis à 1 quelles que soient les valeurs présentes aux entrées *J*, *K* et *CLK*. L'entrée *CLK* ne peut affecter la bascule quand RAU = 0.
- $RAU = 1; RAZ = 0$: La borne RAZ est vraie et Q est *immédiatement* mis à 0 indépendemment des valeurs qu'on peut trouver sur *J*, *K* et *CLK*. L'entrée *CLK* ne peut affecter la bascule quand RAZ = 0.
- $RAU = RAZ = 0$: Cette condition est indésirable puisqu'elle donne lieu à une réponse ambiguë.

Il est important de se rendre compte que les entrées asynchrones sont des niveaux de tension continue (CC). Cela veut dire que si RAU est gardé au niveau 0, la bascule restera dans l'état $Q = 1$ quoiqu'il arrive aux autres entrées. De même, un niveau bas constant sur l'entrée RAZ maintient la bascule dans l'état $Q = 0$. Ainsi les entrées asynchrones peuvent servir à maintenir la bascule dans un certain état pendant un intervalle précis. Le plus souvent, toutefois, on applique momentanément à ces entrées une impulsion pour remettre la bascule à 1 ou à 0.

De nombreuses bascules synchrones fabriquées sous forme de circuits intégrés possèdent les deux entrées asynchrones; d'autres n'ont que l'entrée *RAZ*. Certaines bascules ont des entrées asynchrones qui sont vraies au niveau HAUT plutôt qu'au niveau BAS. Dans le cas de ces dernières, leur symbole est dépourvu des petits ronds devant les entrées synchrones.

Désignation des entrées asynchrones Dans leurs fiches techniques, les fabricants de CI emploient différentes désignations pour les entrées asynchrones, et malheureusement aucune n'a été normalisée. Voici les désignations que vous êtes susceptibles de rencontrer dans les schémas et les fiches techniques anglaises des fabricants. Pour notre part, nous n'utiliserons dans ce livre que *RAU* et *RAZ*.

À partir d'ici, nous désignerons, comme on vient de le dire, les entrées asynchrones au moyen des mentions *RAU* et *RAZ*. Quand ces entrées sont vraies au niveau BAS, et

Entrée asynchrone remise à 1	Entrée asynchrone remise à 0
DC SET	DC CLEAR
PRESET (*PRE*)	CLEAR (*CLR*)
SET	RESET
S_D (direct set)	C_D (direct clear)

c'est généralement le cas, nous utiliserons le trait de surlignement pour indiquer ce fait, par exemple $\overline{RAU}$ et $\overline{RAZ}$.

Même si la plupart des bascules ont au moins une entrée asynchrone, dans de nombreuses applications ces entrées sont inutiles, et c'est pourquoi on les maintient en permanence à leur niveau logique inactif. Dans ce livre, nous ne prendrons pas la peine d'indiquer les entrées asynchrones des bascules à moins qu'elles servent.

EXEMPLE 5-9

La figure 5-31 a) nous montre une bascule déclenchée par un signal dont les entrées asynchrones sont vraies au niveau BAS. Si les entrées J et K sont gardées en permanence au niveau HAUT, déterminez l'onde résultante sur la sortie Q pour les formes d'ondes d'entrée *CLK*, $\overline{RAU}$ et $\overline{RAZ}$. Supposez qu'initialement $Q = 1$.

Solution Au départ, $\overline{RAU}$ et $\overline{RAZ}$ sont dans leur état inactif et n'ont, de ce fait, aucune influence sur Q. Aussi, quand arrive le premier front descendant du signal d'horloge, au point *a*, Q bascule vers son état opposé; rappelez-vous que quand $J = K = 1$, la bascule commute dans l'autre état.

Au point *b*, l'entrée $\overline{RAU}$ passe, la durée d'une impulsion, à son niveau vrai. Ceci met *immédiatement* la bascule à $Q = 1$. Remarquez que $\overline{RAU}$ met la bascule à 1 sans attendre un front descendant de *CLK*. Les entrées asynchrones fonctionnent indépendamment de *CLK*.

Au point *c*, le front descendant de *CLK* a pour effet de faire commuter la bascule dans son état contraire. Voyez $\overline{RAU}$ qui était revenu à son état inactif entre-temps. De la même façon, le front descendant suivant de l'horloge, au point *d*, fait basculer Q vers le niveau HAUT.

Au point *e*, l'entrée $\overline{RAZ}$ est attaquée par une impulsion qui la met temporairement à son niveau vrai BAS; *immédiatement*, la bascule est mise à 0. De nouveau, cela se produit indépendamment du signal d'horloge.

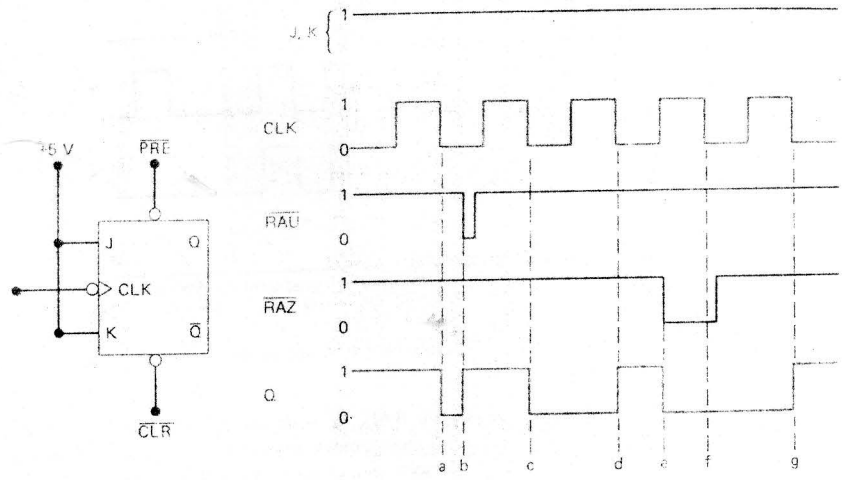
Le front descendant de *CLK*, au point *f*, ne fait pas changer d'état à la bascule, parce que l'entrée $\overline{RAZ}$ est encore active. La valeur BASSE de $\overline{RAZ}$ a priorité sur l'entrée *CLK* et garde $Q = 0$.

Quand le front descendant suivant de *CLK* survient, au point *g*, la bascule commute vers l'état HAUT, puisqu'il n'y a plus d'entrée asynchrone active à cet instant.

Ces étapes sont résumées sur la figure 5-31 b).

QUESTIONS DE RÉVISION

1. Indiquez la différence entre le fonctionnement d'une entrée asynchrone et celui d'une entrée synchrone.
2. Une bascule *D* peut-elle répondre aux états de ses entrées *D* et *CLK* si $\overline{RAU} = 1$?
3. Énumérez les conditions nécessaires pour qu'une bascule J-K déclenchée par un front montant et dont les entrées synchrones sont vraies au niveau BAS, commute vers son état opposé.



a)

Point	Déroulement
a	Basculement synchrone sur FD de CLK
b	Mise à 1 asynchrone par $\overline{RAU} = 0$
c	Basculement synchrone
d	Basculement synchrone
e	Mise à 0 asynchrone par $\overline{RAZ} = 0$
f	$\overline{RAZ}$ a priorité sur le FD de CLK
g	Basculement synchrone

b)

FIGURE 5-31 Formes d'ondes pour l'exemple 5-9; on voit que la bascule synchrone répond à ses entrées asynchrones.

5-10 SYMBOLES IEEE/ANSI

Jusqu'à présent, nous avons décrit les bascules et les éléments de mémoire au moyen des symboles traditionnels, et nous continuerons de les employer dans la plupart de nos diagrammes de circuit. Dans la section que voici, nous allons examiner les nouveaux symboles IEEE/ANSI pour ces dispositifs afin de nous familiariser avec ces derniers.

La figure 5-32 a) nous fait voir le symbole logique d'un simple élément de mémoire *D*. Il s'agit ici du nouveau symbole IEEE/ANSI. On peut voir que l'entrée VALIDATION s'appelle maintenant « C ». Comme nous le verrons, dans la symbologie IEEE/ANSI, on associe la lettre « C » à toute entrée qui *commande* le moment où d'autres entrées auront un effet sur la sortie. Vous savez déjà que l'entrée de VALIDATION autorise la valeur sur la borne *D* à affecter les niveaux sur Q et $\overline{Q}$. Notez que les sorties Q et $\overline{Q}$ sont marquées à l'extérieur du bloc. En plus, observez le petit triangle droit sur $\overline{Q}$ pour signifier que c'est la sortie inversée. C'est la présentation normalisée par IEEE/ANSI. Rappelez-vous que le petit triangle droit est l'équivalent du petit rond des symboles traditionnels.

La figure 5-32 b) illustre le symbole IEEE/ANSI d'un CI précis: le boîtier standard TTL 7475 qui contient quatre éléments de mémoire *D*. Chacun de ces éléments fonctionnent comme nous l'avons décrit précédemment. Ce symbole convient également à tous les CI

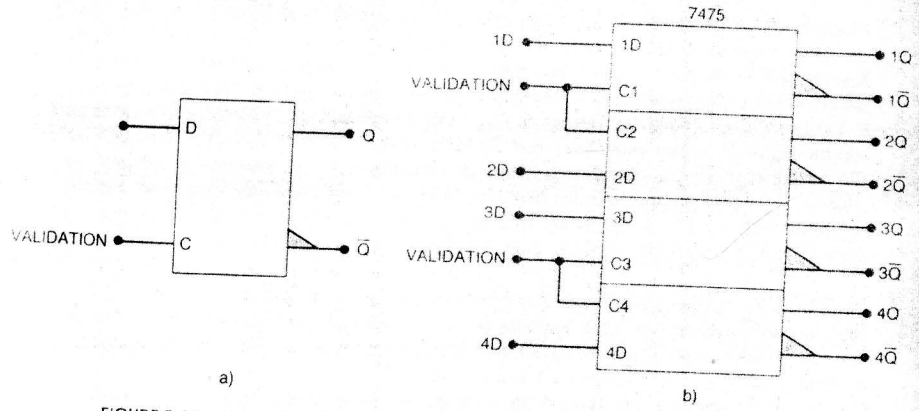


FIGURE 5-32 Symboles IEEE/ANSI pour a) un simple élément de mémoire *D* et b) un CI réel (le 7475 contenant quatre mémoires).

correspondants dans les autres séries TTL et CMOS, par exemple 74LS75, 74L75, 74C75 et 74HC75.

Si on examine le symbole logique du CI 7475, plusieurs points ressortent. D'abord, on peut voir que le tracé du symbole englobe quatre petits rectangles, chacun associé à une mémoire individuelle. Arrêtez-vous également à la façon de désigner les entrées et les sorties de chaque élément de mémoire. Par exemple, l'entrée *D* de la mémoire du haut a pour nom « 1*D* », son entrée de validation s'appelle « *C*1 » et ses sorties sont désignées par 1*Q* et 1$\overline{Q}$. Finalement, remarquez que les deux éléments de mémoire du haut ont une entrée de validation commune, soit *C*1 et *C*2 raccordées en interne et combinées sur une seule broche du boîtier du circuit intégré.

De même les deux mémoires du bas partagent une entrée de validation commune. La figure 5-33 a) montre le symbole logique IEEE/ANSI d'une bascule J-K déclenchée par le front descendant d'un signal et munie d'entrées asynchrones. Le signal d'horloge est identifié par la lettre « C » placée à l'intérieur du rectangle. Notez qu'il y a deux triangles sur l'entrée de l'horloge: un triangle intérieur qui signifie une entrée déclenchée par un front, et le triangle extérieur qui signifie que le front déclencheur est descendant. Les entrées $\overline{RAU}$ et $\overline{RAZ}$ sont vraies au niveau BAS, comme le signalent les deux rectangles droits. Il est intéressant de constater que la norme IEEE/ANSI utilise les mentions « S » et « R » à l'intérieur du rectangle pour désigner aussi la MISE À 1 et la REMISE À 0 asynchrones.

La figure 5-33 b) nous fait voir le symbole logique IEEE/ANSI d'un CI membre de la série 74LS des circuits TTL. Le 74LS112 renferme deux bascules J-K déclenchées par un front descendant dotées d'entrées asynchrones pour la mise à 1 et la remise à 0, chacune de ces bascules est identique à celle décrite à la figure 5-33 a). Remarquez la numérotation des entrées et des sorties, et le fait que les libellés des entrées se limitent à la première bascule. Il faut comprendre que cela signifie que les entrées de la bascule du bas ont le même brochage que celle du haut. Le même symbole sert autant pour le 74LS112 que pour le 74HC112.

La figure 5-34 a) représente le symbole IEEE/ANSI d'une bascule D déclenchée par un front montant qui possède des entrées asynchrones. Dans ce symbole, on ne voit pas de triangle droit sur l'entrée d'horloge, puisque la bascule est synchronisée sur le front montant de l'horloge.

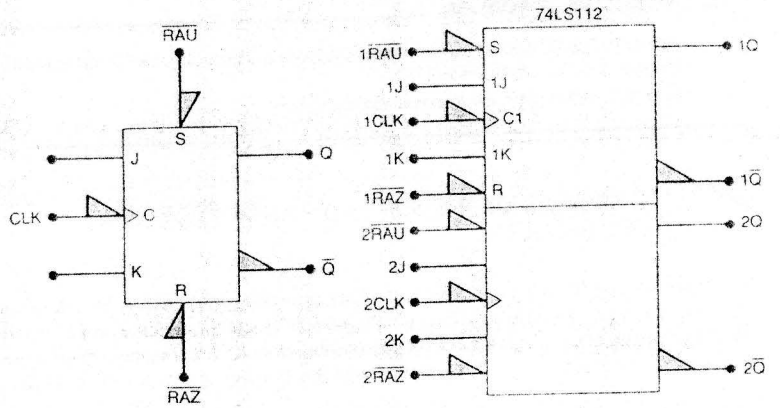
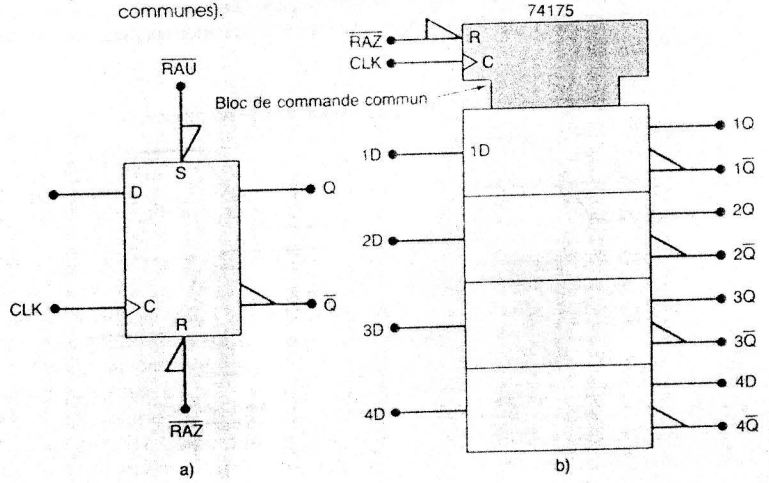


FIGURE 5-33 Symboles IEEE/ANSI pour a) une bascule J-K simple déclenchée par un signal et b) un CI réel (le 74LS112, deux bascules J-K déclenchées par un front descendant).

La partie b) de la figure 5-34 reproduit le CI 74175, qui renferme quatre bascules D partageant une entrée *CLK* commune, ainsi qu'une entrée $\overline{RAZ}$. Les bascules ne disposent pas d'entrée $\overline{RAU}$. Ce symbole renferme un rectangle distinct pour chaque bascule, de même qu'un **bloc de commande commun**, le rectangle encoché du dessus. Ce bloc est utilisé chaque fois qu'avec ce CI on a des entrées communes avec plus d'un élément de la puce. Dans ce cas du 74175, les entrée partagées par les quatre bascules D du boîtier sont le signal d'horloge *CLK* et l'entrée $\overline{RAZ}$. Cela signifie qu'un front montant de l'horloge amène sur chaque sortie *Q* le niveau se trouvant sur l'entrée *D*; en outre, une tension BASSE appliquée sur $\overline{RAZ}$ a pour effet de mettre à zéro toutes les sorties *Q* de la puce.

FIGURE 5-34 Symboles IEEE/ANSI pour a) une bascule D simple déclenchée par un signal et b) un CI réel (le 74175, quatre bascules avec horloge et remise à zéro communes).



QUESTIONS DE RÉVISION

1. Expliquez la signification des deux triangles de forme différente que l'on retrouve sur l'entrée d'horloge dans les symboles IEEE/ANSI.
2. Dites à quoi sert le bloc de commande commun.

5-11 CONSIDÉRATIONS SUR LA SYNCHRONISATION DES BASCULES

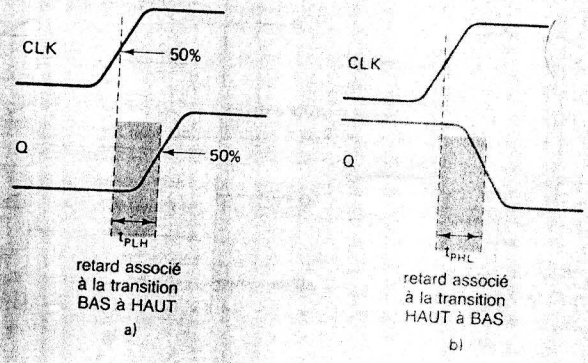
Les fabricants de bascules intégrées spécifient quelques paramètres et caractéristiques de synchronisation importants qu'il importe de prendre en considération avant d'installer une bascule dans un circuit pratique. Nous passons en revue ici les plus importants et donnons quelques exemples de bascules CI réelles tirés des familles logiques TTL et CMOS.

Temps de stabilisation et temps de maintien Nous avons déjà étudié ces deux paramètres à la section 5-4; rappelons que ce sont des temps qu'il faut respecter pour obtenir un déclenchement fiable de la bascule. La fiche technique du fabricant de CI indique toujours les valeurs minimales de t_S et t_M.

Retards de propagation Chaque fois qu'un signal doit changer l'état d'une bascule, on observe un retard entre le moment où le signal est appliqué et le moment où le changement apparaît à la sortie. Une illustration des retards de propagation affectant la réponse à un front montant de signal d'horloge nous est fournie à la figure 5-35. Notez que ces retards sont mesurés entre les points à mi-hauteur (50 %) des formes d'ondes d'entrée et de sortie. Les mêmes genres de retards se produisent en réponse à des signaux placés sur les entrées asynchrones (*RAU* et *RAZ*). Sur les fiches techniques des fabricants, on trouve généralement les retards de propagation affectant la réponse à toutes les entrées, ainsi que les valeurs maximales de t_{PLH} et t_{PHL}.

Les bascules modernes dans les CI ont des retards de propagation qui vont entre quelques nanosecondes et environ 1 μs. Les valeurs de t_{PLH} et t_{PHL} ne sont pas égales et augmentent proportionnellement avec le nombre d'étages logiques pilotés par la sortie Q. Les retards de propagation dans les bascules jouent un rôle important dans certaines situations étudiées plus loin.

FIGURE 5-35 Retards de propagation dans une bascule.



Fréquence de synchronisation maximale, f_{MAX} Il s'agit ici de la fréquence la plus haute que peut avoir le signal d'horloge et qui assure encore un déclenchement fiable de la bascule. La limite f_{MAX} varie d'une bascule à l'autre et même entre bascules ayant le même numéro. Par exemple, si le fabricant d'une puce de bascule J-K 7470 teste plusieurs de ces bascules et trouve des valeurs de f_{MAX} variant entre 20 et 35 MHz, il impose alors comme f_{MAX} la valeur *minimale*, soit 20 MHz. Ceci peut vous sembler quelque peu surprenant, mais si vous réfléchissez bien vous constaterez que la position du fabricant est celle-ci: il ne peut garantir que les bascules 7470 que vous installerez dans votre circuit fonctionneront au-delà de 20 MHz; bien que plusieurs le pourraient. Donc, ce qui est certain c'est qu'en bas de 20 MHz, elles fonctionneront toutes comme prévu.

Durée à 1 et durée à 0 du signal d'horloge Le fabricant spécifie également, d'une part, la durée *minimale* pendant laquelle le signal d'horloge doit demeurer à 0 avant de passer à 1, durée souvent appelée $t_W(L)$, et d'autre part, la durée minimale pendant laquelle le signal d'horloge doit demeurer à 1 avant de revenir à 0, durée souvent appelée $t_W(H)$. Ces durées sont définies à la figure 5-36. Pour que les déclenchements se fassent de façon fiable, il faut toujours respecter ces durées minimales. Prenez note que ces durées sont mesurées aux points de demi-hauteur des fronts du signal.

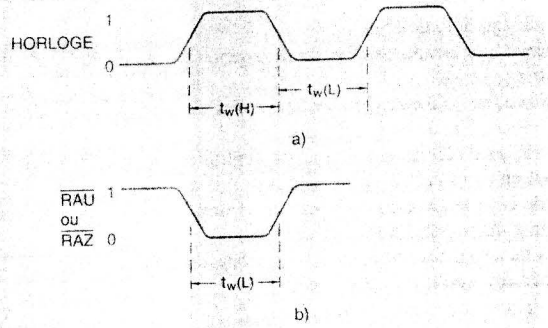


FIGURE 5-36 a) Durée à 1 et durée à 0 du signal d'horloge; b) largeur de l'impulsion asynchrone.

Largeur de l'impulsion asynchrone au niveau vrai Le fabricant inclut aussi dans sa fiche technique la durée *minimale* pendant laquelle il faut garder une entrée *RAU* ou *RAZ* dans son état vrai pour vraiment forcer à 1 ou à 0 la bascule. La figure 5-36 b) montre $t_W(L)$ pour des entrées asynchrones vraies au niveau BAS.

Temps de transition du signal d'horloge Pour obtenir un déclenchement fiable, il faut que les temps de transition (temps de montée et temps de descente) de la forme d'onde du signal d'horloge soient très courts. Des temps de transition trop longs peuvent être la cause de bascules qui se déclenchent n'importe quand, voire pas du tout. Habituellement, les fabricants ne précisent pas un temps de transition maximal pour chaque bascule CI. Ils donnent plutôt une exigence de nature générale pour tous les CI d'une certaine famille. Par exemple, les temps de transition sont habituellement 50 ns pour les dispositifs TTL et 200 ns pour les CMOS. Ces exigences varient légèrement d'un fabricant à l'autre et entre les diverses sous-familles des grandes familles logiques TTL et CMOS.

CI réels À titre d'illustration de ces divers temps, voyons ces paramètres pour quelques bascules intégrées réelles, notamment pour les boîtiers:

- **7474** Deux bascules D déclenchées par un signal d'horloge (famille TTL).
- **74LS112** Deux bascules J-K déclenchées par un signal d'horloge (famille TTL).
- **4013B** Deux bascules D déclenchées par un signal d'horloge (famille OS).
- **74HC112** Deux bascules J-K déclenchées par un signal d'horloge (famille CMOS).

Le tableau 5-1 compare les divers temps de synchronisation de ces bascules tels qu'on les trouve dans les documents techniques des fabricants. Toutes les valeurs de ce tableau sont les valeurs *minimales*, sauf les retards de propagation qui sont les valeurs *maximales*.

L'examen de ce tableau nous amène à faire les constatations suivantes:

1. Toutes les bascules ont des temps de maintien très petits, ce qui est caractéristique de la plupart des bascules modernes déclenchées par un signal d'horloge.
2. La série 74HC des dispositifs CMOS a des paramètres de synchronisation qui se comparent à ceux des dispositifs TTL. Par contre, la série 4000 est beaucoup plus lente que la série 74HC.

EXEMPLE 5-10

En utilisant le tableau 5-1, déterminez:

a) Le temps nécessaire pour que Q passe à 1 quand arrive un FM à l'entrée *CLK* d'un 7474. Supposez $Q = 0$.

b) Le temps nécessaire pour que Q passe à 0 en réponse à un signal $\overline{RAZ}$ sur un 74HC112. Supposez $Q = 1$.

c) L'impulsion la plus étroite que l'on peut appliquer à une entrée *RAZ* d'une bascule 74LS112 tout en étant certain que Q est bien mis à zéro.

d) La bascule de ce tableau dont les entrées de commande doivent rester *stables* après la transition vraie du signal d'horloge.

e) Pour quelles bascules les entrées de commande doivent être stables un minimum de temps avant le front actif de l'horloge?

Solution

a) Le FM fait passer Q de 0 à 1. Le retard entre *CLK* et Q est d'après le tableau $t_{PLH} = 25$ ns pour le 7474.

TABLEAU 5-1 TEMPS DE SYNCHRONISATION DES BASCULES (EN NANOSECONDES)

	TTL		CMOS	
	7474	74LS112	4013B	74HC112
t_S	20	20	60	25
t_H	5	0	0	0
t_{PHL} — de *CLK* à Q	40	24	200	31
t_{PLH} — de *CKL* à Q	25	16	200	31
t_{PHL} — de $\overline{RAZ}$ à Q	40	24	225	41
t_{PLH} — de $\overline{RAU}$ à Q	25	16	225	41
$t_W(L)$ — durée à 0 de *CLK*	37	15	100	25
$t_W(H)$ — durée à 1 de *CLK*	30	20	100	25
$t_W(L)$ — à $\overline{RAU}$ ou $\overline{RAZ}$	30	15	60	25
f_{MAX} — en MHz	15	30	5	20

b) Pour le 74HC112, il faut d'après le tableau $t_{PHL} = 41$ ns pour que Q bascule de 1 à 0 en réponse à l'application d'un signal sur *RAZ*.

c) Pour le 74LS112, l'impulsion la plus étroite applicable à l'entrée *RAZ* est $t_W(L) = 15$ ns.

d) La seule bascule du tableau 5-1 dont le temps de maintien est non nul est dans le boîtier 7474.

e) Toutes les bascules ont, dans leur fiche technique, un temps de stabilisation non nul.

QUESTIONS DE RÉVISION

1. Quels paramètres de synchronisation d'une bascule sont une indication du délai de réponse de la sortie Q à une valeur d'entrée.
2. *Vrai ou faux:* Une bascule dont la f_{MAX} nominale est 25 MHz peut être déclenchée en toute confiance par tout signal d'horloge dont la fréquence est inférieure à 25 MHz.

5-12 PROBLÈME DE SYNCHRONISATION POSSIBLE DANS LES CIRCUITS À BASCULES

Dans nombre de circuits numériques, la sortie d'une bascule est couplée directement, ou par l'intermédiaire de portes logiques, à l'entrée d'une autre bascule, qui sont toutes deux commandées par le même signal d'horloge. Ceci peut conduire à un problème de synchronisation. Examinez à la figure 5-37 une situation type, dans laquelle la sortie Q_1 est connectée à l'entrée J de Q_2 et où les deux bascules sont déclenchées par le même signal sur leur entrée *CLK*.

Voici la description de ce problème de synchronisation possible: étant donnée que Q_1 change à l'instant du front descendant de l'horloge, l'entrée J_2 de Q_2 change au moment de ce même front. Ceci peut se traduire par une réponse imprévisible de la part Q_2.

Supposons qu'au départ $Q_1 = 1$ et $Q_2 = 0$. C'est ainsi que la bascule Q_1 a $J_1 = K_1 = 1$ et Q_2 a $J_2 = Q_1 = 1$, $K_2 = 0$ avant le front descendant de l'horloge. À l'instant de ce front, Q_1 bascule vers son état BAS, mais n'est pas vraiment à 0 avant le délai de propagation t_{PHL}. Le même front descendant commande également le passage de Q_2 dans l'état HAUT à la condition que t_{PHL} soit plus grand que le paramètre du temps de maintien, t_M. Si cette condition n'est pas satisfaite, la réponse de Q_2 est imprévisible.

Heureusement, toutes les bascules modernes déclenchées par un signal ont un temps de maintien égale ou inférieur à 5 ns; plusieurs ont $t_M = 0$, c'est-à-dire n'ont pas besoin d'un temps de maintien. Pour ces bascules, des situations comme celles de la figure 5-37 ne posent pas de risque.

À moins d'indication contraire, dans tous les circuits à bascules que nous vous présenteront dans ce livre, nous supposons que le temps de maintien de la bascule est suffisamment court pour qu'elle réagisse correctement en suivant la règle que voici:

La sortie d'une bascule passe dans l'état imposé par les niveaux logiques actifs sur ses entrées de commande synchrones juste avant la transition active du signal d'horloge.

Si nous appliquons cette règle à la figure 5-37, on apprend que Q_2 passe à l'état imposé par la condition $J_2 = 1$, $K_2 = 0$ existant juste avant l'arrivée du front descendant de l'horloge. Le fait que J_2 change en réponse au passage du même front n'a aucun effet.

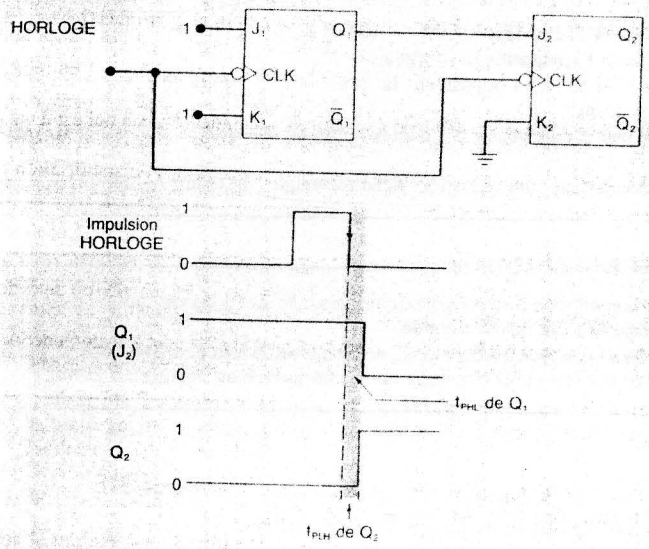


FIGURE 5-37 Q_2 répond correctement au niveau présent sur Q_1 avant le front descendant du signal d'horloge si le temps de maintien de Q_2, t_M, est inférieur au retard de propagation dans Q_1.

EXEMPLE 5-11

Déterminez la sortie Q pour la bascule J-K déclenchée par un front descendant pour les formes d'ondes d'entrée illustrées à la figure 5-38. Supposez qu'initialement $t_M = 0$ et $Q = 0$.

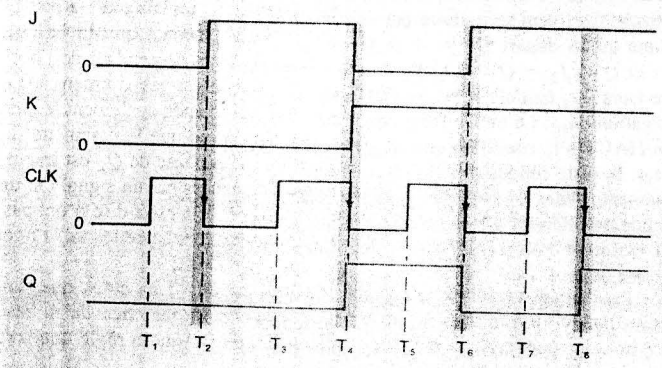


FIGURE 5-38 Exemple 5-11.

Solution La bascule réagit seulement aux instants T_2, T_4, T_6 et T_8. En T_2, Q répond à la condition $J = K = 0$ présente juste avant. En T_4, Q répond à la condition $J = 1$, $K = 0$ présente juste avant. En T_6, Q répond à la condition $J = 0$, $K = 1$ présente juste avant. Et enfin, en T_8, Q répond à la condition $J = K = 1$.

5-13 BASCULES MAÎTRE-ESCLAVE

Avant la mise au point des bascules déclenchées par un signal ayant un temps de maintien faible ou nul, on réglait fréquemment les problèmes de synchronisation, comme celui illustré à la figure 5-37, en recourant à une catégorie de bascules appelée bascules *maître-esclave*. En réalité, une bascule maître-esclave est formée de deux bascules: une bascule maître et une bascule esclave. Au moment du front montant de l'horloge, les niveaux sur les entrées de commande (D, J, K) ont pour rôle de déterminer la sortie de la bascule maître. Quand le signal CLK passe au niveau BAS, l'état de la bascule maître est transféré à la bascule esclave, dont les sorties sont Q et $\overline{Q}$. Ainsi, Q et $\overline{Q}$ changent juste après le front descendant de l'horloge. Ces bascules maître-esclave ont un fonctionnement sensiblement analogue à celui d une bascule déclenchée par un front descendant, sauf pour un inconvénient important: les entrées de commande doivent rester stables pendant que CLK est au niveau HAUT, sans quoi le comportement est imprévisible. Ce problème propre aux bascules maître-esclave a été résolu par la mise au point d'une version améliorée appelée *maître-esclave à verrouillage de donnée*.

La bascule maître-esclave est dorénavant désuette, mais on peut en retrouver encore dans l'équipement moins moderne. Des exemples de tels éléments sont, en technologie TTL version standard, les boîtiers 7473, 7476 et 74107, et en version à verrouillage de donnée, les boîtiers 74110 et 74111. Les technologies de CI les plus récentes (74LS, 74AS, 74ALS, 74HC, 74HCT) ne contiennent plus de bascules maître-esclave dans leurs séries. En fait, les séries 74LS76 et 74LS107 renferment des bascules déclenchées par un signal, même si leur série 74 standard correspondante contient des bascules maître-esclave.

Pour les besoins de la chose, si vous vous retrouvez devant une bascule maître-esclave dans une pièce d'équipement, vous pourrez toujours l'analyser comme s'il s'agissait d'une bascule déclenchée par un front descendant.

5-14 APPLICATIONS DES BASCULES

Nous avons déjà vu comment les simples bascules en NON-ET ou en NI peuvent nous permettre de réaliser des interrupteurs antirebond (exemple 5-2) et une mémoire (exemple 5-4). Ces bascules asynchrones élémentaires n'entrent toutefois pas dans une gamme d'applications très large. Par contre, les bascules synchrones sont des outils très polyvalents grâce auxquels un concepteur peut envisager un grand nombre d'applications. Nous allons dans les sections suivantes présenter certaines applications parmi les plus courantes, applications que nous décrirons dans les chapitres suivants de façon exhaustive.

5-15 SYNCHRONISATION DES BASCULES

Presque tous les systèmes numériques fonctionnent surtout en synchronisme avec certaines transitions d'un signal d'horloge. Cependant, il arrive souvent qu'un signal extérieur ne soit pas synchronisé avec le signal d'horloge, donc qu'il soit asynchrone. Les signaux asynchrones sont souvent lancés par un opérateur qui ouvre ou ferme un interrrupteur à un moment quelconque par rapport au signal d'horloge. Ce manque de synchronisme peut parfois produire des résultats imprévisibles et indésirables. Voici un exemple où on montre comment une bascule peut servir à synchroniser l'action d'une entrée asynchrone.

EXEMPLE 5-12

À la figure 5-39 a) un signal A est lancé par un interrupteur antirebond qu'actionne un opérateur (on a construit à l'exemple 5-2 un interrupteur antirebond). Le signal A passe au niveau HAUT quand l'opérateur ouvre l'interrupteur et passe au niveau BAS quand il ferme l'interrupteur. Ce signal sert à commander le transfert du signal d'horloge à la borne de sortie X; la porte ET est transparente au signal d'horloge quand A est à 1.

Le problème posé par ce circuit vient du fait que A peut changer de valeur en tout temps par rapport au signal d'horloge puisque l'opérateur peut ouvrir ou fermer l'interrupteur quand bon lui semble. On peut donc retrouver sur X des parties d'impulsion d'horloge, si A change d'état pendant que CLK est au niveau HAUT; c'est ce que nous montre la figure 5-39 b).

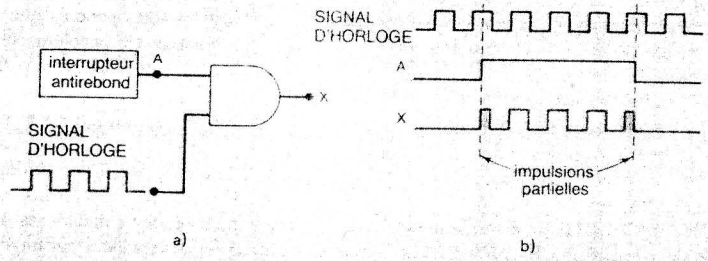
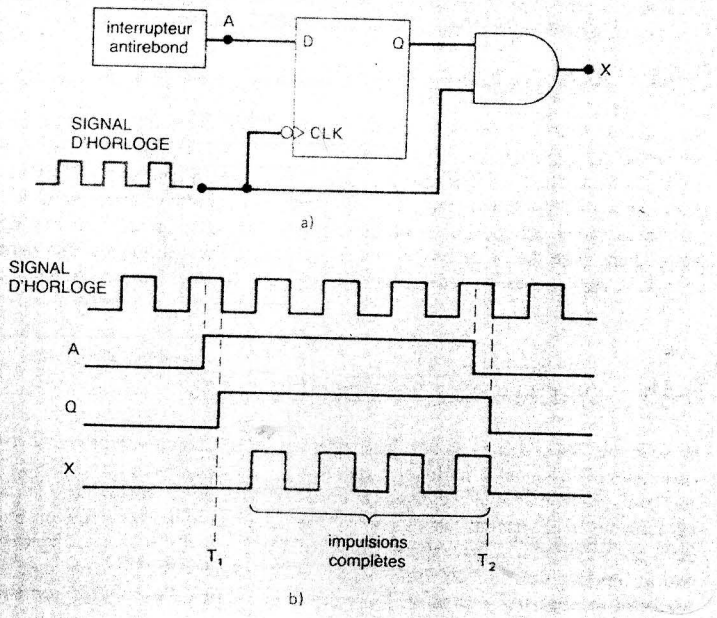


FIGURE 5-39 Un signal asynchrone comme A peut créer des impulsions partielles sur X.

FIGURE 5-40 Utilisation d'une bascule D pour synchroniser la validation de la porte ET avec les FD du signal d'horloge



Ce genre de sortie est indésirable la plupart du temps; il faut donc prévenir l'apparition d'impulsions d'horloge partielles sur X. Une solution pour y parvenir est le circuit de la figure 5-40 a). Décrivez la méthode matérialisée par ce circuit pour résoudre notre problème et tracez la forme d'onde de X pour les conditions de la figure 5-39 b).

Solution Le signal A est raccordé à l'entrée D de la bascule Q, synchronisée avec le FD du signal d'horloge. Ainsi, quand A passe à 1, Q ne devient pas 1 tant que le FD suivant du signal d'horloge n'est pas arrivé, c'est-à-dire pas avant T_1. Cette valeur haute sur Q va alors autoriser la porte ET à laisser passer les impulsions suivantes *complètes* jusqu'à X, comme le montre la figure 5-40 b).

Quand A revient à 0, Q, lui, ne revient pas à 0 avant le FD suivant, soit pas avant T_2. De cette façon la porte ET ne bloque pas les impulsions d'horloge avant que l'impulsion se terminant à T_2 ne soit apparue sur X. On ne retrouve alors en sortie sur X que des impulsions complètes.

5-16 DÉTECTION D'UNE SÉQUENCE D'ENTRÉE

La sortie de certains circuits ne réagit qu'à une succession particulière de signaux sur les entrées. Un tel comportement est impossible à réaliser avec des circuits combinatoires, il faut pour cela exploiter la possibilité de rétention des bascules.

Par exemple, une porte ET peut servir à déterminer que deux entrées, A et B, sont toutes les deux au niveau HAUT. Cependant, une telle porte est incapable de réagir différemment selon que c'est A ou B qui passe à 1 en premier. Maintenant, supposons que l'on veut avoir une sortie au niveau HAUT seulement si A passe à 1 avant que B passe à 1. Un façon d'y parvenir est de construire le circuit de la figure 5-41 a).

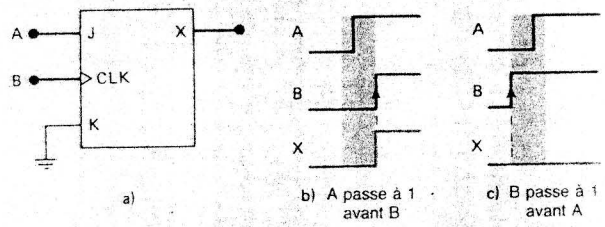


FIGURE 5-41 Utilisation d'un bascule synchrone J-K pour détecter une séquence particulière de signaux d'entrée.

Les formes d'ondes des parties b) et c) de cette figure montrent que X sera au niveau HAUT seulement si A passe à 1 avant B. Il en est ainsi parce que A doit être à 1 pour que X change d'état au moment du FM du signal B.

Pour qu'un tel circuit fonctionne convenablement, il faut que A passe à 1 avant B un certain laps de temps égal au temps de stabilisation prescrit pour cette bascule.

5-17 STOCKAGE ET TRANSFERT DE DONNÉES

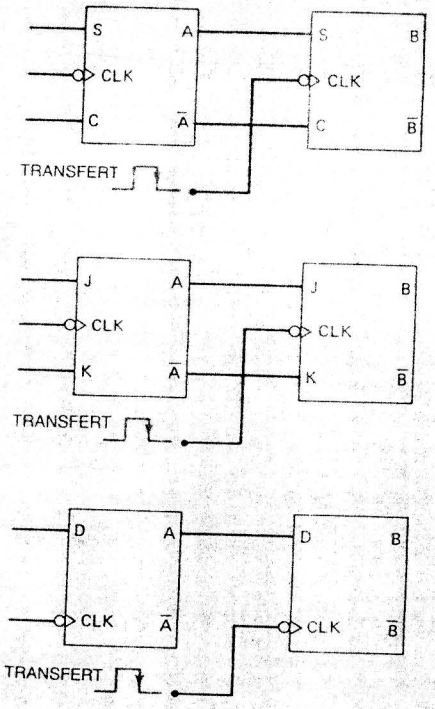
C'est comme éléments de mémorisation de données ou d'information que les bascules sont les plus utilisées. Les données stockées peuvent être des valeurs chiffrées comme des

nombres binaires, des caractères codés DCB. Le stockage des données a généralement lieu dans des groupes de bascules, appelés **registres**.

La manipulation la plus fréquente qu'on fait subir aux données conservées dans des bascules ou des registres est le **transfert**. Par transfert, on entend l'échange de données d'un registre à un autre. La figure 5-42 nous fait voir comment s'effectue le transfert de données entre deux bascules synchrones pour les trois types de bascules que l'on connaît: la S-C, la J-K et la D. Dans chaque cas, la valeur logique en mémoire dans la bascule *A* est transférée à la bascule *B* à l'arrivée du FD de l'impulsion TRANSFERT. Après cette impulsion, on retrouve la même sortie en *B* qu'en *A*.

Les opérations de transfert de la figure 5-42 sont dites **synchrones**, parce qu'elles mettent en jeu les entrées synchrones et *CLK*. Il est également possible d'échanger des données en utilisant les entrées asynchrones des bascules. La figure 5-43 représente un **transfert asynchrone** dans lequel on utilise les entrées *RAU* et *RAZ* de la bascule. Dans ce cas-ci, les entrées asynchrones sont vraies au niveau BAS, comme on peut le déduire des petits ronds qui se trouvent sur ces entrées. Quand la ligne VALIDATION TRANSFERT est maintenue au niveau BAS, les sorties des deux portes NON-ET sont gardées à 1, de sorte que les entrées asynchrones n'ont pas d'effet sur la bascule. Quand cette ligne passe au niveau HAUT, une des deux portes NON-ET passe à 0; cette porte est déterminée par l'état des sorties *A* et $\overline{A}$. Cette valeur 0 a pour effet de mettre à 1 ou à 0 la bascule *B* en vue de lui communiquer le même état qui se trouve dans *A*. Ce transfert asynchrone s'effectue indépendamment des entrées synchrones et *CLK* des bascules. On qualifie parfois

FIGURE 5-42 Transfert synchrone pour trois types de bascules.



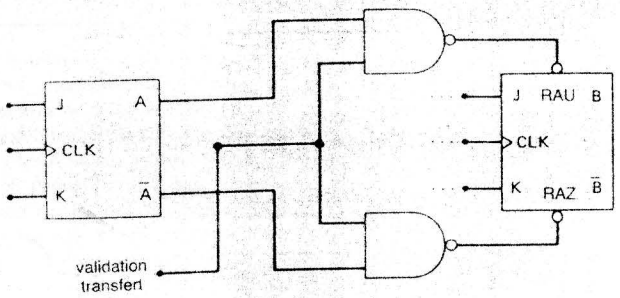


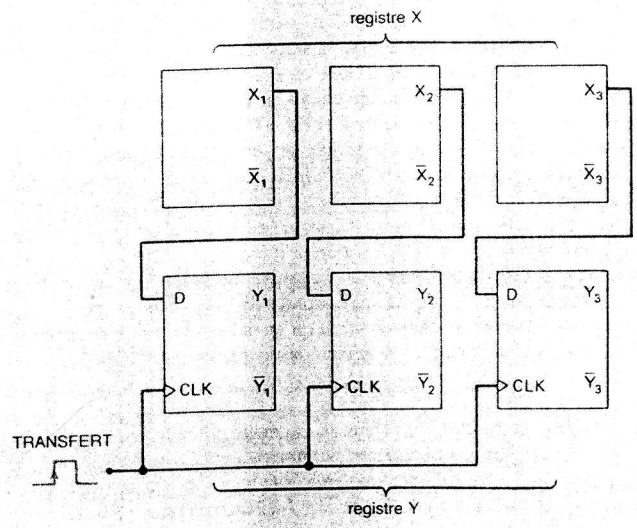
FIGURE 5-43 Transfert asynchrone.

le transfert asynchrone de **transfert direct**, puisque les données sont placées dans la bascule réceptrice même si les entrées synchrones sont actives.

Transfert de données en parallèle On peut voir à la figure 5-44 un transfert de données entre un certain registre et un autre constitué de bascules *D*. Le registre *X* est formé de trois bascules X_1, X_2, X_3, alors que le registre *Y* est formé des trois bascules Y_1, Y_2, Y_3. Le transfert des données du registre *X* au registre *Y* a lieu de façon synchrone. On dit aussi que c'est un *transfert en parallèle* parce que le contenu de X_1, X_2, X_3 est transféré *simultanément* dans Y_1, Y_2, Y_3. Si le transfert avait eu lieu en *série*, le contenu du registre *X* aurait été communiqué un bit à la fois au registre *Y*. Ce dernier type de transfert fait d'ailleurs l'objet de la section suivante.

Il importe de bien saisir le fait que le transfert parallèle ne modifie pas le contenu du registre qui fournit les données. Par exemple, sur la figure 5-44, si $X_1X_2X_3 = 101$ et $Y_1Y_2Y_3 = 011$ avant l'apparition de l'impulsion TRANSFERT, et bien, la valeur dans les deux registres après l'impulsion TRANSFERT est 101.

FIGURE 5-44 Transfert en parallèle du contenu du registre *X* dans le registre *Y*.



QUESTIONS DE RÉVISION

1. *Vrai ou faux:* On utilise l'entrée *CLK* dans les transferts de données asynchrones.
2. Nommez le type de bascule qui convient le mieux au transfert synchrone, puisqu'il exige moins d'interconnexions entre les bascules.
3. Si les bascules de la figure 5-44 étaient des J-K, dites combien il faudrait d'interconnexions pour raccorder une bascule du registre *X* à son homologue du registre *Y*.
4. *Vrai ou faux:* Il faut moins de circuits pour réaliser un transfert synchrone qu'un transfert asynchrone.

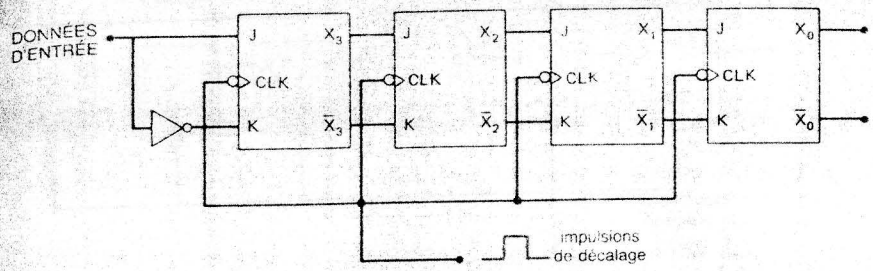
5-18 TRANSFERT EN SÉRIE: REGISTRES À DÉCALAGE

Avant d'aborder la description du transfert de données en série, il serait bon d'examiner le montage de base d'un *registre à décalage*. Un registre à décalage est un ensemble de bascules câblées de manière à ce que les nombres mémorisés dans les bascules soient déportés d'une bascule à la suivante à chaque impulsion d'horloge. Sans vous en doutez, vous avez déjà vu à l'œuvre des registres à décalage dans certains matériels électroniques, comme les calculatrices électroniques, dans lesquelles les chiffres affichés se déplacent vers la gauche à mesure qu'on entre de nouveaux chiffres. C'est la même action qui se déroule dans un registre à décalage. La figure 5-45 a) nous montre quatre bascules J-K câblées de manière à constituer un registre à décalage de 4 bits. Notez que les bascules sont raccordées pour que la sortie de X_3 soit transférée dans X_2, celle de X_2 dans X_1 et celle de X_1 dans X_0. Un tel agencement fait donc en sorte qu'à l'arrivée d'une impulsion de décalage chaque bascule prend l'état mémorisé avant l'impulsion dans la bascule immédiatement à gauche. Pour sa part X_3 mémorise la valeur définie par les conditions présentes sur ses entrées *J* et *K* à l'arrivée de l'impulsion de décalage. Pour le moment, présumons que les entrées *J* et *K* sont alimentées par la forme d'onde DONNÉES D'ENTRÉE de la figure 5-45 b). Nous allons supposer également que toutes les bascules sont à l'état 0 avant l'application des impulsions de décalage.

Les formes d'ondes de la figure 5-45 b) démontrent comment l'application d'une succession d'impulsions de décalage déplace les données d'entrée de gauche à droite d'une bascule à l'autre. À l'arrivée du premier FD, en T_1, les entrées de chacune des bascules X_2, X_1 et X_0 sont $J = 1$ et $K = 0$, compte tenu de l'état des bascules se trouvant à leur gauche. Donc à cet instant, X_3 passe au niveau HAUT tandis que les autres bascules restent à 0. À l'instant T_2, moment du deuxième FD, les entrées de X_3 sont $J = 0$ et $K = 1$, puisque ce sont les valeurs fournies par DONNÉES D'ENTRÉE. Les entrées de la bascule X_2 sont à $J = 1$ et $K = 0$ parce qu'il y a un 1 à la sortie de X_3. Les entrées de X_1 et X_0 sont toujours $J = 1$ et $K = 0$. Ainsi à T_2 seule la bascule X_2 passe au niveau HAUT, tandis que X_3 revient à 0 et que X_1 et X_0 restent à 0.

Par un raisonnement analogue, on arrive à trouver de quelle façon changent les formes d'ondes aux instants T_3 et T_4. Notez qu'à chaque FD des impulsions de décalage, les sorties des bascules prennent la valeur de sortie qu'avait la bascule immédiatement à gauche *avant* le FD. Évidemment, X_3 prend la valeur fournie par la ligne DONNÉES D'ENTRÉE avant le FD.

Temps de maintien à respecter Dans le montage d'un registre à décalage, il est impérieux que les bascules aient un temps de maintien excessivement faible, du fait que parfois les entrées *J* et *K* changent presque en même temps que le passage de la transition du signal d'horloge. Ainsi, la sortie X_3 bascule de 1 à 0 quand passe le FD en T_2, provoquant



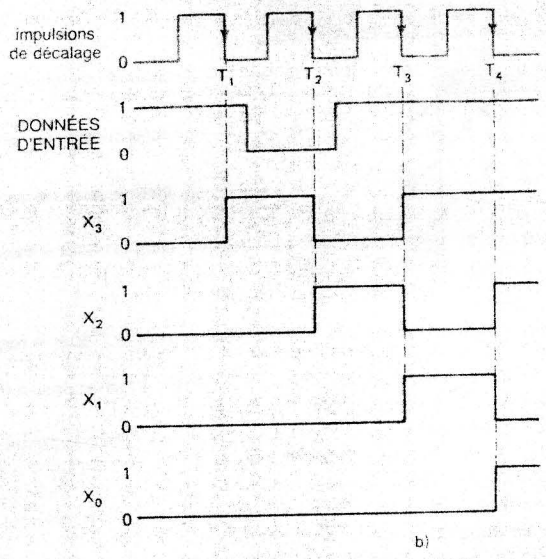


FIGURE 5-45 Registre à décalage de 4 bits.

le changement des sorties *J* et *K* de X_2 en même temps que *CLK* subit son changement. En réalité, le retard de propagation introduit par X_3 fait en sorte que les entrées de *J* et *K* resteront inchangées pendant une courte période après le FD. C'est la raison pour laquelle, il faut absolument que les registres à décalage soient construits avec des bascules déclenchées par un signal d'horloge dont t_M est plus petit que le retard de propagation d'une seule bascule (*CLK* à sortie). La majorité des bascules modernes déclenchées par un signal d'horloge respectent facilement cette dernière exigence.

Transfert registre à registre en série La figure 5-46 représente des registres à décalage de 3 bits raccordés de telle sorte que le contenu du registre est transféré en série (par décalage) dans le registre *Y*. Chacun de ces registres est formé de bascules D, bascules qui exigent moins de connexions que les bascules J-K. Voyez comment X_0, la sortie de la dernière bascule du registre *X*, est connectée à l'entrée Y_2, celle de la première bascule du registre *Y*. Ainsi, les impulsions de décalage déclenchent le transfert d'information selon le schéma suivant: $X_2 \rightarrow X_1 \rightarrow X_0 \rightarrow Y_2 \rightarrow Y_1 \rightarrow Y_0$. La bascule X_2 prend la valeur définie

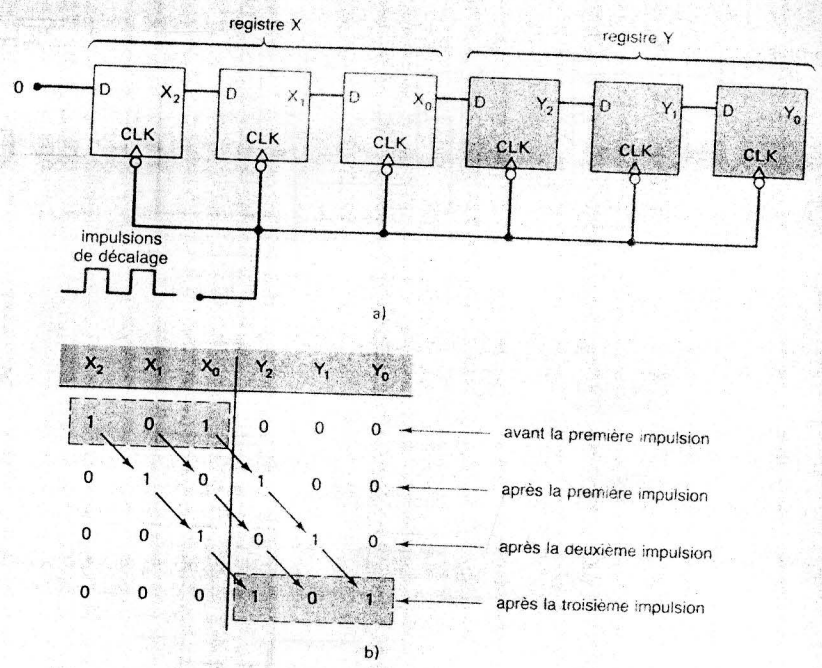


FIGURE 5-46 Transfert en série de l'information du registre X dans le registre Y.

par l'entrée D. Pour le moment, D est gardée au niveau BAS; X_2 passe donc à 0 quand arrive la première impulsion et reste à cette valeur.

Illustrons ce fonctionnement en supposant qu'avant l'arrivée de la première impulsion le contenu de X soit 1 0 1 (c.-à-d. $X_2 = 1, X_1 = 0, X_0 = 1$) et qu'il n'y ait que des zéros dans Y. Reportez-vous au tableau de la figure 5-46 b); voyez comment les états de chaque bascule sont décalés vers la droite à mesure que des impulsions sont appliquées. Voici certains points que nous voulons souligner:

1. Au front descendant de chaque impulsion, les bascules prennent la valeur mémorisée dans la bascule immédiatement à gauche avant l'arrivée de l'impulsion.
2. Après *trois impulsions*, le 1 qui se trouvait dans X_2 est dans Y_2, le 0 qui se trouvait dans X_1 est dans Y_1 et le 1 qui se trouvait dans X_0 est dans Y_0. Le 101 mémorisé dans le registre X a été transmis par décalage dans le registre Y. La valeur se trouvant maintenant dans le registre X est 000; ce registre a perdu sa valeur initiale.
3. Il a fallu *trois* impulsions pour transférer *trois* bits d'information.

EXEMPLE 5-13

Supposons que les registres X et Y contiennent au départ les mêmes données. Quel est le contenu de chaque bascule après le passage de la sixième impulsion d'horloge?

Solution Si vous poursuivez le processus appliqué au montage de la figure 5-46 b) pendant trois autres impulsions, vous vous apercevrez que toutes les bascules sont dans l'état 0 après la sixième impulsion. Une autre façon d'arriver au même résultat est de faire l'analyse suivante: le 0 constant à l'entrée D et X_2 fournit un nouveau 0 à chaque impulsion, de sorte qu'après six impulsions toutes les bascules sont garnies de 0.

Comparaison transfert en parallèle–transfert en série Dans le transfert en parallèle, toute l'information est communiquée en une seule fois à l'arrivée d'une *seule* impulsion de commande de transfert (figure 5-44), et cela quel que soit le nombre de bits à transférer. Dans le transfert en série, décrit sur la figure 5-46, l'échange de N bits d'information réclame N impulsions d'horloge (3 impulsions pour 3 bits, 4 impulsions pour 4 bits...). De toute évidence, le transfert en parallèle est nettement plus rapide que le transfert en série effectué au moyen de registres à décalage.

Dans le transfert en parallèle, chaque sortie de la bascule de départ est raccordée à l'entrée de la bascule correspondante du registre d'arrivée. Dans le transfert en série, seule la dernière bascule du registre de départ est raccordée au registre d'arrivée. En général, quand on fonctionne en parallèle, il faut plus d'interconnexions que lorsqu'on fonctionne en série. Cette remarque est d'autant plus vraie qu'on transfère un plus grand nombre de bits. C'est une considération non négligeable quand le registre émetteur est éloigné du registre récepteur, puisque le nombre de lignes nécessaires à la transmission ne sera pas le même selon que l'on fonctionne en série ou en parallèle.

C'est l'application particulière et les spécifications du système qui conditionnent le choix entre transfert en série et transfert en parallèle. Souvent, dans un même système, on profite du meilleur de ces deux genres de transfert: dans une partie du système on exploite la *rapidité* du transfert en parallèle et dans une autre on mise sur la *simplicité et l'économie* du transfert en série.

QUESTIONS DE RÉVISION

1. *Vrai ou faux:* Le transfert en parallèle est la méthode la plus rapide pour transmettre de l'information d'un registre à un autre.
2. Dites quel est le principal avantage du transfert en série sur le transfert en parallèle.
3. Reportez-vous à la figure 5-46. Présumez qu'initialement les contenus des registres sont: $X_2 = 0, X_1 = 1, X_0 = 0, Y_2 = 1, Y_1 = 1, Y_0 = 0$. Présumez aussi que l'entrée D de X_2 est gardée au niveau HAUT. Déterminez la valeur de chacune des sorties des registres après la quatrième impulsion.

5-19 DIVISION DE LA FRÉQUENCE ET COMPTAGE

Pour cette section, reportez-vous à la figure 5-47 a). Chaque bascule a ses entrées J et K au niveau 1; on sait que dans cette condition, la sortie de la bascule change d'état à chaque fois que l'impulsion d'horloge passe du niveau HAUT au niveau BAS. Le train d'impulsions d'horloge est appliqué seulement à l'entrée *CLK* de la bascule X_0. La sortie de cette dernière bascule est amenée à l'entrée *CLK* de la bascule X_1 dont la sortie est, elle-même, amenée à l'entrée *CLK* de la bascule X_2. La figure 5-47 b) montre de quelle façon changent les sorties des bascules à mesure que sont appliquées les impulsions. Voici des points importants qu'il convient de souligner:

1. La bascule X_0 passe à l'état opposé à l'arrivée du front descendant de chaque impulsion d'horloge. Aussi la forme d'onde de sortie de X_0 a une fréquence exactement deux fois plus petite que celle du signal d'horloge.

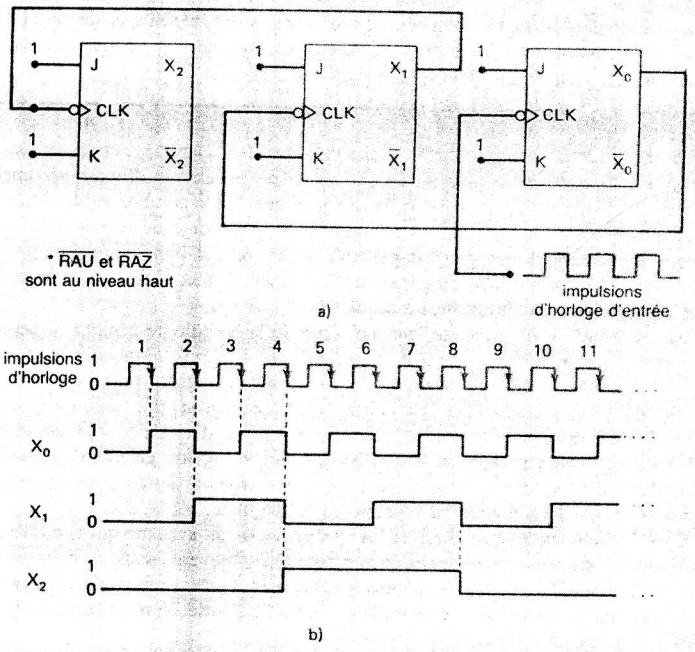


FIGURE 5-47 Bascules J-K câblées de façon à réaliser un compteur binaire de 3 bits (MODULO 8).

2. La bascule X_1 passe à l'état opposé chaque fois que la sortie X_0 effectue la transition de HAUT à BAS. La forme d'onde de X_1 a donc une fréquence exactement égale à la moitié de celle de X_0, donc exactement égale au quart de la fréquence d'horloge.
3. La bascule X_2 passe à l'état opposé chaque fois que X_1 effectue la transition de HAUT à BAS. La forme d'onde de X_2 a une fréquence égale à la moitié de celle de X_1, donc égale au 1/8e de la fréquence d'horloge.
4. Chaque sortie de bascule est une onde carrée (onde ayant un facteur de forme de 50 %).

Comme nous venons de le décrire, chaque bascule divise par deux la fréquence du signal qui alimente son entrée *CLK*. Donc, si on avait ajouté une quatrième bascule aux trois premières, la fréquence de la sortie finale aurait été 1/16e de la fréquence d'horloge; et ainsi de suite pour une cinquième, une sixième... bascule. On peut avec un circuit de ce genre, ayant le nombre de bascules qu'il faut, diviser la fréquence initiale par n'importe quelle puissance de 2. Plus précisément, avec N bascules, on obtient un signal de sortie dont la fréquence est égale à $1/2^N$ de la fréquence d'entrée.

Comptage Le circuit de la figure 5-47, en plus d'être un diviseur de fréquence, est un **compteur binaire**. C'est ce qu'on démontre en examinant la suite des **états** pris par les bascules après le passage de chaque impulsion d'horloge. La figure 5-48 expose les résultats de cet examen sous forme d'un tableau. Mettons que la suite $X_2X_1X_0$ représente un nombre binaire dans lequel X_2 est le chiffre dont le poids est 2^2, X_1, le chiffre dont le poids est

2^2 $\overline{X_2}$	2^1 $\overline{X_1}$	2^0 $\overline{X_0}$	
0	0	0	avant la première impulsion
0	0	1	après la 1e impulsion
0	1	0	après la 2e impulsion
0	1	1	après la 3e impulsion
1	0	0	après la 4e impulsion
1	0	1	après la 5e impulsion
1	1	0	après la 6e impulsion
1	1	1	après la 7e impulsion
0	0	0	après la 8e impulsion revient à 000
0	0	1	après la 9e impulsion
0	1	0	après la 10e impulsion
0	1	1	après la 11e impulsion
.	.	.	. . .
.	.	.	. . .
.	.	.	. . .

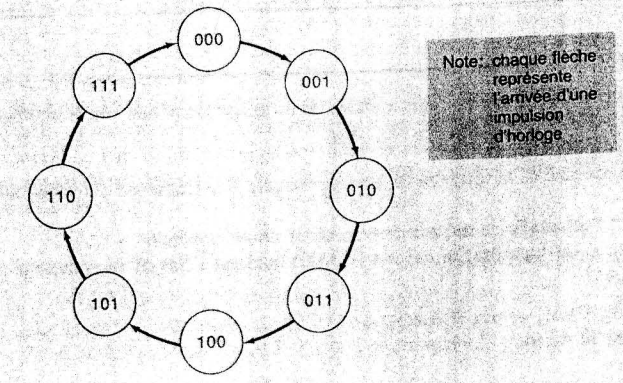
FIGURE 5-48 La suite des états des bascules concorde avec la suite des nombres binaires.

2^1 et X_0, le chiffre dont le poids est 2^0. Les huit premiers états $X_2X_1X_0$ correspondent à la suite des nombres binaires entre 000 et 111. Après la première impulsion, les bascules affichent l'état 001 ($X_2 = 0$, $X_1 = 0$ et $X_0 = 1$) ce qui représente 001_2 (l'équivalent binaire du 1 décimal); après la deuxième impulsion, les bascules affichent 010, soit l'équivalent de 2_{10}; après trois impulsions, $011_2 = 3_{10}$; après quatre impulsions, $100_2 = 4_{10}$; et ainsi de suite jusqu'à sept impulsions, $111_2 = 7_{10}$. À la huitième impulsion, les bascules reviennent à l'état 000, et la suite des nombres binaires recommence pour les impulsions suivantes.

En conclusion, si on établit une correspondance entre les états des bascules et les nombres binaires, ce circuit peut compter jusqu'à 7_{10} avant d'être ramené à 000.

Graphe de transition des états Une autre façon de visualiser le changement des états des bascules après chaque application d'un signal d'horloge est d'utiliser un *graphe de transition des états* comme celui de la figure 5-49. Chaque cercle représente un état possible,

FIGURE 5-49 Graphe de transition qui dénote le changement des états du compteur (ses bascules) après chaque impulsion d'horloge.



soit l'état indiqué par le nombre binaire intérieur. Par exemple, le cercle renfermant le nombre 100 correspond à l'état 100 (soit, $X_2 = 1$, $X_1 = 0$ et $X_3 = 0$).

Les flèches reliant chaque cercle figurent le changement d'état quand arrive une impulsion d'horloge. En examinant un cercle d'état particulier, on peut connaître l'état précédent et l'état suivant. Par exemple, pour l'état 000, on s'aperçoit que le compteur atteint cet état quand le compteur est dans l'état 111 et qu'arrive une impulsion d'horloge. De la même façon, on s'aperçoit que l'état qui suit 000 est toujours l'état 001.

Nous utiliserons les graphes de transition pour nous aider à décrire, analyser et concevoir des compteurs et d'autres types de circuits séquentiels à bascules.

MODULO Le compteur de la figure 5-47 peut prendre 2^3 états différents (000 à 111). On dit que c'est un *compteur MODULO-8*, où MODULO est un nombre indiquant le nombre d'états dénombrés dans la suite des nombres. Si on avait ajouté une quatrième bascule, la suite des états aurait permis de compter de 0000 à 1111, soit jusqu'à 16 états. On aurait alors parlé d'un *compteur MODULO-16*. En généralisant, N bascules raccordées comme dans le montage de la figure 5-47 réalisent un compteur ayant 2^N états, de sorte que c'est un compteur MODULO-2^N. Ce dernier circuit est en mesure de compter jusqu'à $2^N - 1$ avant d'être réinitialisé.

Le MODULO d'un compteur indique de surcroît le chiffre par lequel la fréquence est divisée à la sortie de la dernière bascule. Par exemple, un compteur de 4 bits possède 4 bascules, chacune correspondant à un chiffre binaire; c'est donc aussi un compteur MODULO-2^4 = MODULO-16. Ce compteur peut compter jusqu'à 15 et il peut diviser la fréquence d'entrée par un facteur de 16 (le MODULO).

Le compteur binaire que nous venons de vous présenter ici est rudimentaire. Au Chapitre 7, nous étudierons en détail les compteurs.

EXEMPLE 5-14

Si on prend comme hypothèse que le compteur MODULO-8 de la figure 5-47 est dans l'état 101. Dans quel état (nombre) se retrouve-t-il après 13 impulsions?

Solution Localisez l'état 101 sur le graphe de transition des états. Déplacez-vous sur le graphe des états de 8 positions, vous devriez être revenu sur l'état 101. Continuez encore pendant les cinq états suivants (pour un total de 13), vous devriez vous trouvez sur l'état 010.

Remarquez qu'avec un compteur MODULO-8, il faut parcourir huit états pour effectuer un tour complet du graphe et pour revenir au point de départ.

EXEMPLE 5-15

Soit un compteur formé de six bascules câblées de la même façon que sur le montage de la figure 5-47 (c.-à-d. $X_5, X_4, X_3, X_2, X_1, X_0$).

a) Déterminez son MODULO.
b) Trouvez la fréquence de la sortie de la dernière bascule si la fréquence d'horloge initiale est 1 MHz.
c) Dites quel est l'intervalle de la suite des nombres de ce compteur.
d) Supposez que l'état (nombre) initial est 000000; quel est l'état de ce compteur après 129 impulsions?

Solution

a) MODULO = $2^6 = 64$.

b) La fréquence de la sortie de la dernière bascule est la fréquence d'horloge initiale divisée par le MODULO, soit:

$$f(\text{à } X_5) = \frac{1 \text{ MHz}}{64} = 15.625 \text{ kHz}$$

c) La suite des nombres va de 000000_2 à 111111_2 (0 à 63_{10}), ce qui correspond à 64 états. Notez que ce dernier chiffre est égal au MODULO.

d) Ce compteur étant modulo 64, après 64 impulsions, le compteur est revenu à son état initial. Même chose après 128 impulsions. Donc, après 129 impulsions, l'état du compteur est 000001.

QUESTIONS DE RÉVISION

1. On applique un signal d'horloge de 20 kHz à une bascule dont les entrées J et K sont toutes les deux à 1. Dites quelle est la fréquence de l'onde de sortie de la bascule.
2. Dites combien il faut de bascules pour pouvoir compter de 0 à 255_{10}.
3. Donnez le MODULO du compteur de la question 2.
4. Donnez la fréquence de la sortie de la 8e bascule si le signal d'entrée a une fréquence de 512 kHz.
5. Si l'état initial de ce compteur est 00000000, quel est l'état du compteur après 520 impulsions.

5-20 APPLICATION AU MICRO-ORDINATEUR

Notre étude des circuits numériques en est encore à ses tout débuts, et vous n'avez pas encore appris grand chose des microprocesseurs et des micro-ordinateurs. Malgré cela, vous pouvez arriver à concevoir relativement bien l'utilité des bascules dans une application commandée par microprocesseur, sans connaître tous les détails que nous nous proposons d'aborder avec vous dans les chapitres suivants.

La figure 5-50 représente un microprocesseur (MP) dont les sorties servent à transférer des données binaires vers un registre X, constitué de quatre bascules D désignées X_3, X_2, X_1 et X_0. Un groupe de sorties du MP sert pour le *code d'adresse* et comprend les huit sorties A_{15}, A_{14}, A_{13}, A_{12}, A_{11}, A_{10}, A_9, A_8. Dans la plupart des MP, il y a 16 sorties d'adresse disponibles, qui ne sont, cependant, pas toutes utilisées simultanément. Un second groupe de sorties du microprocesseur comprend quatre *lignes de données* D_3, D_2, D_1, D_0. Dans les MP courants, c'est plutôt huit lignes de donnée que l'on retrouve. La dernière sortie du MP est le signal d'horloge.

Rappelez-vous que le microprocesseur est l'unité centrale de traitement d'un micro-ordinateur, et son principal rôle est d'exécuter les instructions stockées dans la mémoire de la machine. Une des instructions possibles est celle qui commande au MP de transférer un nombre binaire contenu dans un registre du MP vers un registre extérieur X. Les étapes de l'exécution de cette instruction dans le MP sont:

1. Placer le nombre binaire sur les sorties de donnée D_3 à D_0.
2. Placer le code d'adresse qui sélectionne le registre X, auquel est destiné la donnée, sur les lignes d'adresse A_{15} à A_8.
3. Après la stabilisation des sorties de donnée et d'adresse, le MP génère l'impulsion d'horloge CP qui active le registre et complète le transfert de la donnée dans X.

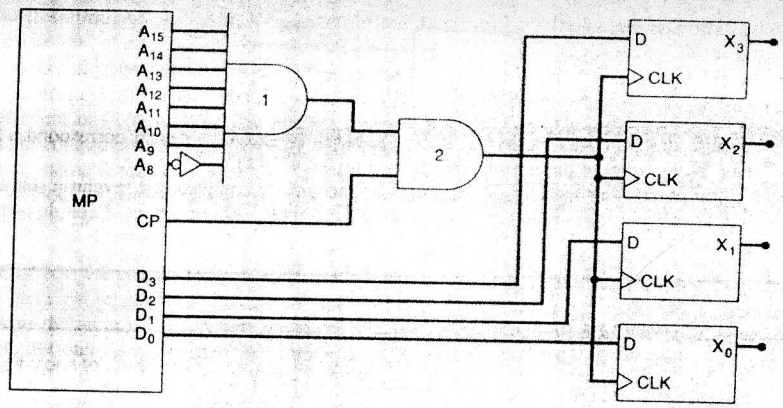


FIGURE 5-50 Illustration d'un microprocesseur transférant des données binaires dans un registre extérieur.

Les bascules du registre X peuvent servir à différents usages. Par exemple, dans les cas où le MP vient de terminer un calcul dont les résultas doivent être affichés, chaque bascule pourrait être raccordée à une DEL, de sorte que la donnée transférée est immédiatement visible. Autre exemple: chaque bascule pourrait, selon le code binaire envoyé par le MP, contrôler la mise en marche ou l'arrêt de dispositifs électromagnétiques, comme des moteurs ou des bobines. Évidemment, le MP étant lui-même sous la gouverne du programme qu'il est chargé d'exécuter.

EXEMPLE 5-16

a) Quel code d'adresse le MP doit-il générer pour que les données aboutissent dans le registre X (figure 5-50)?

b) Supposons que $X_3 - X_0 = 0110$, $A_{15} - A_8 = 11111111$ et $D_3 - D_0 = 1011$. Quelle est la valeur dans X après le signal d'horloge?

Solution

a) Pour que la donnée placée sur les lignes de donnée passe dans le registre X, il faut que l'impulsion d'horloge puisse franchir la porte ET 2 et parvenir, ainsi, jusqu'aux bornes d'horloge des bascules. Cela ne survient que si l'entrée supérieure de la porte ET 2 est un niveau HAUT. Il faut donc que toutes les entrées de la porte ET 1 soient au niveau HAUT, ou plus précisément que A_{15} à A_9 soient à 1 et que $A_8 = 0$. Ainsi, le code d'adresse qui autorise le transfert des données jusque dans X est 11111110.

b) Si $A_8 = 1$, la sortie BASSE de ET 1 bloque le signal CP qui ne peut franchir la porte ET 2, et les bascules ne sont pas activées par l'horloge. Donc, le contenu du registre X demeure inchangé.

5-21 DISPOSITIFS À BASCULES DE SCHMITT

Un circuit à *bascule de Schmitt* ne figure pas dans la catégorie des bascules proprement dite, même s'il réalise une certaine mémorisation, bien utile dans certaines situations spéciales. L'une d'elle est représentée à la figure 5-51 a). On peut y voir une INVERSEUR

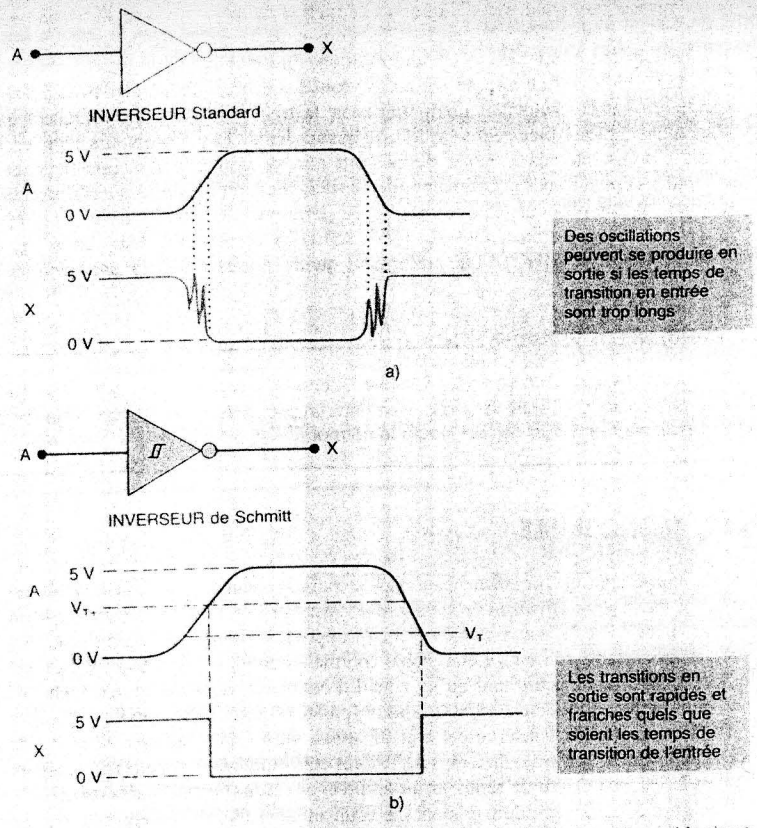


FIGURE 5-51 a) Si les temps de transition sont trop longs, la sortie d'un élément logique standard peut osciller ou changer de façon erratique; b) un élément logique ayant une entrée traitée par une bascule de Schmitt a une sortie dont les transitions sont rapides et bien franches.

piloté par une entrée logique dont les temps de transition sont relativement longs. Quand ces temps de transitions dépassent certaines valeurs admissibles (fonctions de la famille logique utilisée), les sorties des portes logiques et des INVERSEURS peuvent être le siège d'oscillations quand le signal d'entrée se trouve dans l'intervalle indéterminé. Les mêmes conditions d'entrée peuvent se traduire par un déclenchement erratique des bascules.

Un élément qui dispose d'une entrée de type à bascule de Schmitt peut être piloté par des signaux d'entrée lents et produire quand même des changements de niveau sans oscillations. La sortie bascule généralement très rapidement (généralement en 10 ns) indépendamment des caractéristiques du signal d'entrée. La figure 5-51 b) montre un INVERSEUR de Schmitt, de même que sa réponse à une entrée qui change lentement de niveau.

Si vous examinez les formes d'ondes de la figure 5-51 b) vous constaterez que la sortie ne passe pas de la valeur HAUTE à la valeur BASSE avant que l'entrée n'ait dépassé la tension **seuil de montée** V_{T+}. Une fois que la sortie est au niveau BAS, elle le demeure même si l'entrée revient sous la tension seuil V_{T+} (c'est ici qu'intervient la mémorisation);

la sortie ne revient au niveau HAUT que si le signal d'entrée passe sous la tension **seuil de descente** V_{T-}. Les valeurs de ces deux tensions sont fonctions de la famille logique, mais, comme on s'en doute bien V_{T-} sera toujours inférieure à V_{T+}.

L'INVERSEUR de Schmitt, de même que tous les dispositifs à bascules de Schmitt, sont représentés par des symboles distinctifs (voir la figure 5-51 b)), qui indiquent qu'il s'agit d'éléments aptes à bien répondre à des signaux lents. Les concepteurs logiciens se servent des circuits intégrés à entrée de Schmitt afin de convertir les signaux lents en signaux rapides et bien francs qui peuvent piloter adéquatement des entrées de CI standard.

Le marché offre plusieurs CI avec des entrées à bascule de Schmitt: les boîtiers 7414, 74LS14 et 74HC14 contiennent tous six INVERSEURS de Schmitt. Autre exemple, les puces 7413, 74LS13 et 74HC13 qui renferment quatre portes NON-ET avec entrées de Schmitt.

QUESTIONS DE RÉVISION

1. Que se passe-t-il quand un signal qui bascule lentement attaque un CI logique standard?
2. Quelle différence y a-t-il entre le fonctionnement d'un dispositif logique à bascule de Schmitt et le fonctionnement d'un dispositif logique normal.

5-22 MONOSTABLE

Un circuit numérique qui possède beaucoup de points communs avec la bascule est le *monostable*, dit aussi *multivibrateur monostable* parce qu'il n'a qu'un état stable. Comme la bascule, il est doté de deux sorties Q et $\overline{Q}$ opposées. Toutefois, contrairement à la bascule, le monostable n'a qu'*un* état stable (normalement $Q = 0$ et $\overline{Q} = 1$), état dans lequel le monostable demeure tant qu'un signal d'entrée ne vient pas le déclencher. Dès que le monostable est déclenché, ses sorties passent dans l'état opposé ($Q = 1$ et $\overline{Q} = 0$). Le monostable conserve cet état dit **quasi stable** pendant une durée t_p, fixée généralement par une constante de temps RC. Après cette durée t_p, les sorties du monostable reviennent à leur état stable de repos jusqu'au prochain signal de déclenchement.

La figure 5-52 a) reproduit le symbole d'un monostable. La valeur t_p est toujours indiquée d'une manière quelconque sur le symbole du monostable. Dans les circuits pratiques t_p varie de quelques nanosecondes à quelques dizaines de secondes. La valeur précise de t_p est déterminée par la valeur des composants externes R_T et C_T.

Il existe dans le commerce deux types de monostables intégrés: le monostable **nonredéclenchable** et le monostable **redéclenchable**.

Monostable nonredéclenchable Les formes d'ondes de la figure 5-52 nous font voir le fonctionnement d'un monostable nonredéclenchable qui réagit aux fronts montants sur son entrée T. Voici quelques constatations importantes:

1. Les FM, aux instants *a*, *b*, *c* et *e*, vont mettre le monostable dans son état quasi stable pendant une durée t_p après laquelle il revient dans son état stable.
2. Les FM aux instants *d* et *f* n'ont aucun effet sur le monostable puisqu'il se trouve déjà dans son état quasi stable. Le monostable doit revenir dans son état stable avant d'être redéclenché.
3. La durée de l'impulsion de sortie est toujours constante quelle que soit la durée des impulsions d'entrée. Comme nous l'avons déjà dit, t_p est fonction essentiellement de R_T et C_T et des circuits internes du monostable. Un monostable type a un t_p égal à $0{,}7R_TC_T$.

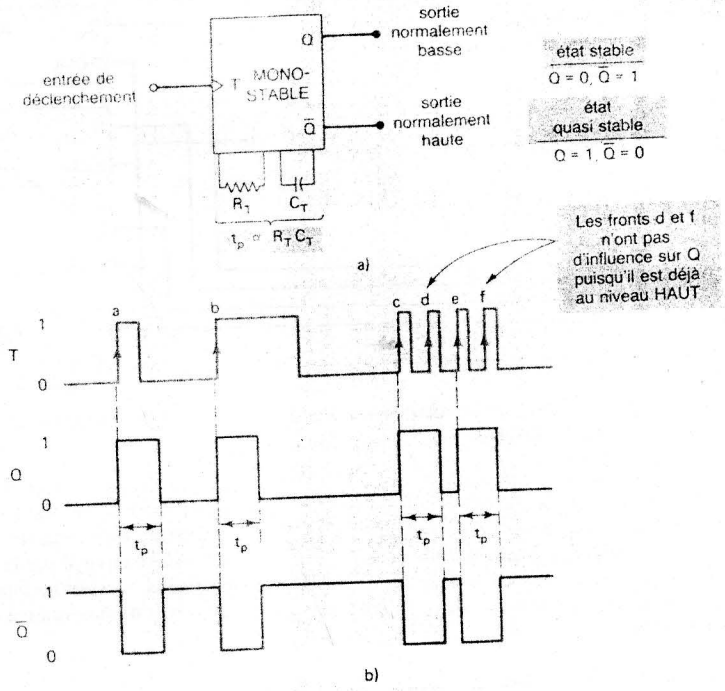


FIGURE 5-52 Symbole d'un monostable et formes d'ondes types.

Le monostable nonredéclenchable en version CI le plus courant est le CI TTL 74121; cette puce contient un monostable nonredéclenchable à trois entrées qui peuvent servir à le déclencher au passage d'un FM ou bien d'un FD.

Monostable redéclenchable Le monostable redéclenchable fonctionne presque comme le monostable nonredéclenchable à une différence près: *il peut être redéclenché quand il se trouve dans son état quasi stable, et dans ce cas une nouvelle impulsion d'une durée t_p est amorcée*. La figure 5-53 a) compare les réponses des deux types de monostables, tous les deux ont une t_p de 2 ms. Examinons ces formes d'ondes.

Pour ces deux types, après un premier déclenchement à l'instant $t = 1$ ms, la sortie garde la valeur HAUTE pour une durée de 2 ms avant de revenir au niveau BAS. La seconde impulsion de déclenchement à $t = 5$ ms place les monostables dans leur état HAUT. La troisième impulsion de déclenchement, à $t = 6$ ms, ne change rien dans le cas du monostable nonredéclenchable, puisque ce dernier est déjà dans son état quasi stable. Par contre, pour ce qui est du monostable redéclenchable, cette impulsion le *relance* et initie un nouvel intervalle de 2 ms dans l'état HAUT. Donc, ce dernier restera au niveau 1 pendant 2 ms après la troisième impulsion de déclenchement.

En effet, le monostable redéclenchable amorce un nouvel interval t_p chaque fois qu'arrive une impulsion de déclenchement, et cela indifféremment de l'état actuel de la sortie Q. Donc, si un train d'impulsions est appliqué à un rythme suffisamment rapide, il pourra arriver que le monostable soit toujours redéclenché avant la fin de l'intervalle t_p et que Q reste continuellement dans l'état HAUT. C'est ce que fait voir la figure 5-53 b),

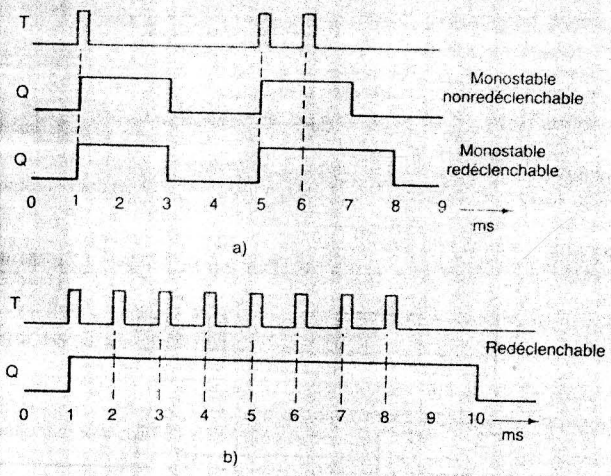
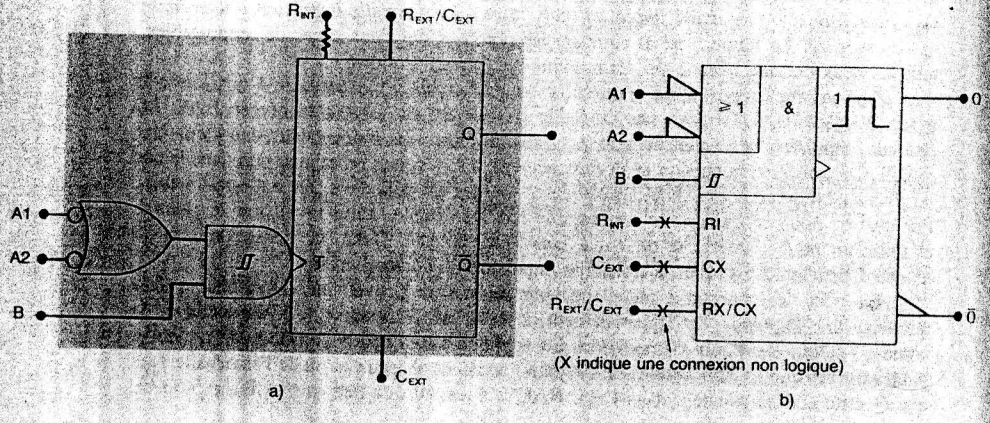


FIGURE 5-53 a) Comparaison des réponses de deux monostables ayant une $t_p = 2$ ms, un nonredéclenchable et un redéclenchable; b) un monostable redéclenchable amorce un nouvel intervalle t_p à chaque arrivée d'une impulsion de déclenchement.

sur laquelle on voit huit impulsions de déclenchement espacées de 1 ms. Q ne revient donc à son niveau stable que 2 ms après la dernière impulsion.

Dispositifs réels Les fabricants offrent plusieurs monostables en circuits intégrés, aussi bien en version nonredéclenchable qu'en version redéclenchable. Ainsi, on retrouve les boîtiers de monostables nonredéclenchables 74121 et 74L121; et les boîtiers 74221, 74LS221 et 74HC221 qui contiennent deux monostables nonredéclenchables. De même les puces 74122 et 74L122 contiennent un monostable redéclenchable, tandis que les 74123, 74LS123 et 74HC123 en renferment deux.

FIGURE 5-54 Symboles logiques du monostable nonredéclenchable 74121: a) traditionnel; b) IEEE/ANSI.

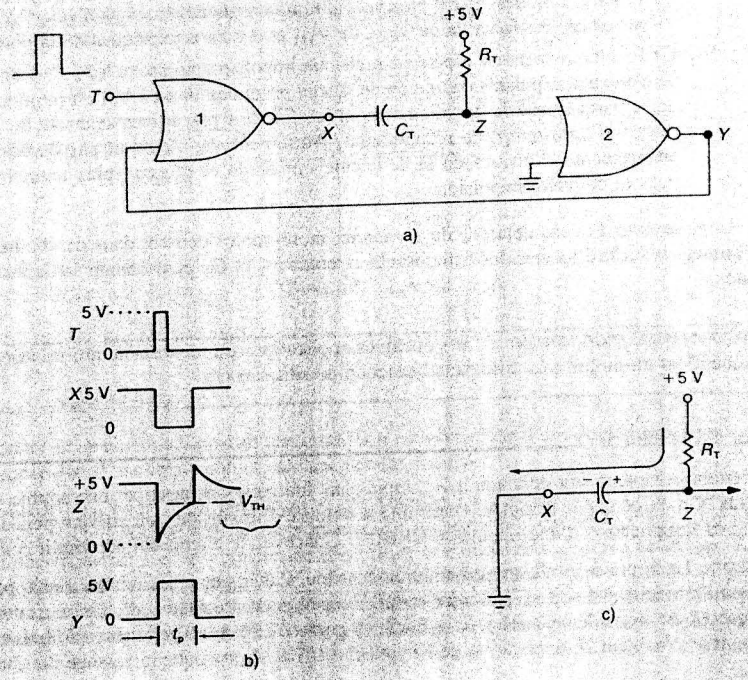


La figure 5-54 a) illustre le symbole traditionnel d'un monostable nonredéclenchable 74121. Remarquez la présence de portes logiques internes qui permettent aux entrées A_1, A_2 et B de déclencher le monostable de différentes manières. L'entrée B est une entrée à bascule de Schmitt, grâce à laquelle même des entrées variant très lentement peuvent déclencher en toute confiance le monostable. Les broches appelées R_{INT}, R_{EXT}/C_{EXT} et C_{EXT} servent pour le raccordement d'une résistance et d'un condensateur extérieurs qui fixent la durée de l'impulsion de sortie. La figure 5-54 b) représente le symbole IEEE/ANSI du monostable nonredéclenchable 74121. Notez la façon dont on représente dans ce symbole les portes logiques. Remarquez en outre la petite forme d'impulsion précédée d'un petit 1. Ce signe annonce que le dispositif est un monostable nonredéclenchable. Le symbole IEEE/ANSI d'un monostable redéclenchable n'aura pas ce petit 1 devant la forme d'impulsion.

Fonctionnement d'un circuit monostable Les monostables, tout comme les bascules, existent dans le commerce sous forme intégrée, de sorte qu'on n'a besoin que d'une résistance et d'un condensateur pour produire une impulsion de sortie ayant la durée voulue. Nous croyons toutefois qu'il peut être instructif d'analyser succinctement le fonctionnement d'un circuit monostable réel. La figure 5-55 montre un tel circuit formé de deux portes NI. Dans l'analyse suivante, reportez-vous aux formes d'ondes tracées sur cette figure.

1. Supposez que l'entrée de déclenchement soit initialement à 0 et que le circuit soit en régime établi.

FIGURE 5-55 a) Circuit d'un monostable; b) formes d'ondes; c) chemin de charge C_T.



2. Une tension HAUTE (+ 5 V) est appliquée sur l'entrée Z de la porte NI 2 au travers d'une résistance R_T. Cette tension HAUTE place un 0 à la sortie Y.
3. Quand $T = Y = 0$, la sortie X de la porte NI 1 est à 1. Comme X et Z sont tous les deux à 1, la différence de potentiel aux bornes du condensateur C_T est faible ou nulle.
4. Quand l'entrée T reçoit une impulsion de niveau HAUT, X passe immédiatement à 0. Étant donné que C_T ne peut accumuler sa charge instantanément, il faut que la tension à Z passe aussi au niveau BAS. Le condensateur C_T joue momentanément le rôle d'un court-circuit entre Z et X.
5. Le 0 à Z fait passer Y au niveau HAUT, et ce niveau HAUT garde X à 0 même si T revient à 0.
6. Le condensateur commence alors à se charger au travers de R_T, comme le montre la figure 5-54 c). N'oubliez pas que X est maintenant presque au potentiel de la masse. Quand C_T se charge, la tension en Z monte.
7. Quand la tension en Z atteint une valeur suffisante, la porte NI est validée et Y revient à 0. Cette valeur porte le nom de tension seuil, V_{TH}. Quand Y passe à 0, X passe à 1.
8. Le front montant qui se produit en X est couplé à Z par le biais de C_T, ce qui fait augmenter la tension en Z. Cette tension revient ainsi graduellement à son niveau en régime établi pendant que C_T se décharge.
9. Ce qu'il faut retenir ici c'est que la durée t_p des impulsions X et Y est directement proportionnelle à la constante de temps $R_T C_T$. Donc t_p est égal au nombre de constantes de temps nécessaire pour que C_T se charge jusqu'à V_{TH}. Dans les conditions normales, t_p est plus petit qu'une constante de temps $R_T C_T$. Cette valeur exacte est fonction de V_{CC}, de V_{TH} et des caractéristiques de la porte.
10. Ce fonctionnement est répété pour chaque impulsion qui arrive à T; c'est donc dire que des impulsions d'une durée t_p sont produites en X et Y. Il est important de remarquer que la deuxième impulsion ne doit pas arriver avant la fin de l'intervalle du temps de récupération. Autrement, C_T ne serait pas déchargé entièrement à l'arrivée de la deuxième impulsion et t_p serait plus court que pour la première impulsion.

Comme nous l'avons déjà dit, de nombreux monostables existent dans des boîtiers CI auxquels il suffit d'ajouter des composants externes R_T et C_T pour obtenir la durée t_p voulue.

Applications du monostable Voici quelques applications du monostable qui montrent bien que c'est un circuit très utile dans beaucoup de situations.

EXEMPLE 5-17

Un certain compteur doit dénombrer les impulsions fournies sans arrêt par une source de signaux. On exige que le compteur détermine le nombre d'impulsions qui arrivent pendant 1 s après l'application d'une impulsion DÉBUT. Concevez le circuit qui accomplit cela.

Solution La figure 5-56 illustre une solution possible. Les impulsions initiales sont appliquées à l'entrée d'une porte ET; l'autre entrée de cette porte reçoit le Q d'un monostable. Puisque Q est normalement au niveau BAS, la porte ET reste à 0 et aucune impulsion n'atteint le compteur. L'application de l'impulsion DÉBUT provoque le passage de Q au niveau HAUT, et Q reste à ce niveau un temps $t_p = 1$ s. Lorsque $Q = 1$, la sortie de la porte ET laisse arriver les impulsions jusqu'au compteur qui les dénombre. Après 1 s, le monostable revient à 0, et le compteur ne reçoit plus d'autres impulsions.

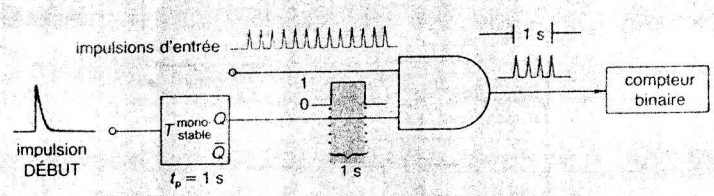


FIGURE 5-56 Exemple 5-17.

EXEMPLE 5-18

Concevez un circuit qui génère une impulsion positive d'une durée de 50 μs quand deux tensions logiques, A et B, sont toutes les deux aux niveaux HAUT. Employez un monostable qui réagit aux fronts descendants.

Solution Une solution possible est le circuit de la figure 5-57. La sortie de la porte NON-ET est normalement au niveau HAUT. Quand A et B passent tous les deux à 1, cette sortie descend à 0. Le front descendant sur cette sortie déclenche le monostable qui fournit en C une impulsion positive d'une durée de 50 μs.

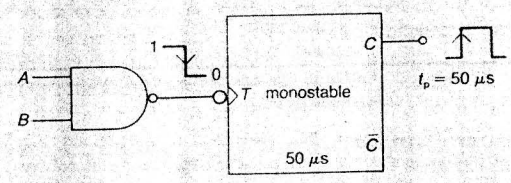


FIGURE 5-57 Exemple 5-18.

EXEMPLE 5-19

Concevez un circuit qui produit une impulsion positive d'une durée de 100 μs, 20 μs après l'apparition d'une impulsion de déclenchement positive.

Solution La figure 5-58 nous montre une solution possible où on emploie deux monostables déclenchés par des fronts montants. Le monostable A est amorçé par l'impulsion de déclenchement, ce qui génère sur la sortie $\overline{A}$ une impulsion négative de 20 μs. Après ces 20 μs, la sortie $\overline{A}$ revient à 1; ce front montant déclenche le monostable B, qui génère sur sa sortie B une impulsion positive de 100 μs. On voit donc que le premier monostable introduit la temporisation de 20 μs et que le deuxième produit l'impulsion de sortie recherchée.

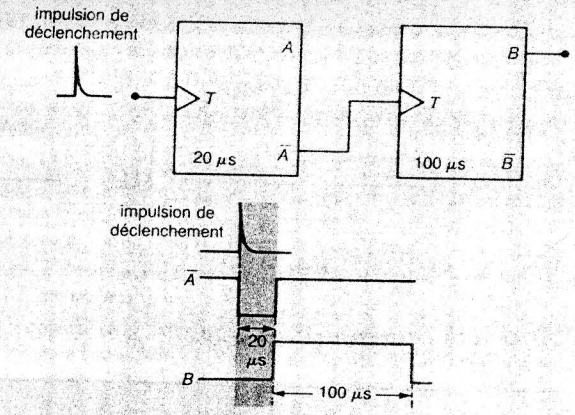


FIGURE 5-58 Exemple 5-19.

QUESTIONS DE RÉVISION

1. En l'absence d'une impulsion de déclenchement, dites quel est l'état de sortie du monostable.
2. *Vrai ou faux:* Quand un monostable nonredéclenchable reçoit une impulsion et qu'il est dans l'état quasi stable, sa sortie ne change pas.
3. Dites ce qui détermine la valeur de t_p d'un monostable.
4. Donnez la différence de fonctionnement entre un monostable redéclenchable et un monostable nonredéclenchable.

5-23 MINUTERIE CI 555

La minuterie NE/SE 555 est un boîtier de CI très polyvalent qui se comporte soit comme un multivibrateur *monostable* (un état stable), soit comme un multivibrateur *astable* (sans état stable). Ce boîtier a les caractéristiques suivantes:

a) Possède une large gamme de tensions d'alimentation (5-15 V).
b) Pilote sans problème des TTL sous une alimentation de 5 V.
c) A une grande capacité de courant de sortie (200 mA).
d) Manifeste une temporisation très stable malgré des variations de la tension d'alimentation et de la température.

Les circuits internes de plusieurs étages de la minuterie 555 sont constitués de transistors bipolaires; on peut voir les étages les plus importants de la minuterie 555 dans le schéma fonctionnel reproduit à la figure 5-59. Ce montage est formé de deux comparateurs, d'une bascule S-C, d'un diviseur de tension qui fournit les tensions de référence aux comparateurs, d'un transistor bipolaire de décharge (Q_1) et d'un tampon en totem pour l'étage de sortie. Ce boîtier possède huit broches: une broche pour V_{CC} et une pour la masse et six autres broches dont les rôles sont décrits ci-après.

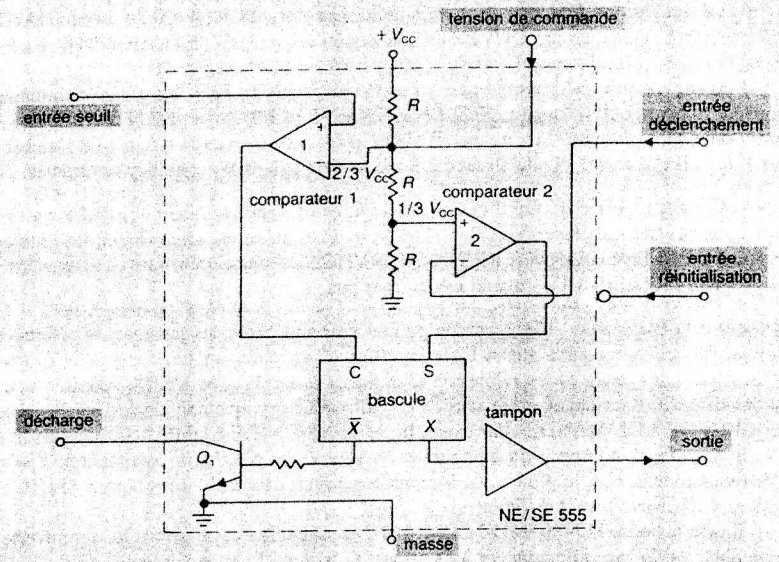


FIGURE 5-59 Schéma fonctionnel de la minuterie 555

SORTIE Cette broche est la ligne de sortie d'un tampon en totem qui peut absorber ou fournir un courant de 200 mA. Elle peut attaquer directement des TTL quand $V_{CC} = 5$ V. Le niveau reporté sur SORTIE (HAUT ou BAS) est celui de la borne X de la bascule.

DÉCHARGE Cette broche est raccordée à un transistor à collecteur ouvert attaqué par la sortie $\bar{X}$ de la bascule. Quand $\bar{X}$ est au niveau BAS, le transistor Q_1 est BLOQUÉ, et on peut considérer la broche DÉCHARGE comme un circuit ouvert. Quand $\bar{X}$ est au niveau HAUT, Q_1 est CONDUCTEUR et la broche DÉCHARGE joue maintenant le rôle d'une résistance très petite par rapport à la masse. Cette broche, comme nous le verrons, est raccordée aux condensateurs utilisés dans les diverses applications du 555, et c'est par cette broche que les condensateurs se déchargent.

SEUIL Cette broche excite l'entrée « + » du comparateur 1. L'entrée « − » de ce même comparateur est raccordée à une tension de référence fixée à 2/3 de V_{CC} au moyen d'un diviseur de tension formé de trois résistances égales. Quand la tension d'entrée SEUIL est $> 2/3\ V_{CC}$, la sortie de ce comparateur passe au niveau HAUT pour remettre à zéro la bascule et mettre ainsi X à 0 et $\bar{X}$ à 1. Quand SEUIL est $< 2/3\ V_{CC}$, la sortie du comparateur se trouve au niveau BAS, et cela n'affecte pas la bascule.

La broche SEUIL sert donc à mettre la SORTIE au niveau BAS et la borne DÉCHARGE en court-circuit. Notez que la broche SEUIL ne permet pas de mettre à 1 la bascule; c'est la borne DÉCLENCHEMENT qui sert à mettre à 1 la bascule.

DÉCLENCHEMENT Cette broche attaque l'entrée « − » du comparateur 2. L'entrée « + » de ce même comparateur est reliée à une tension de référence fixée à 1/3 V_{CC}. Tant que l'entrée DÉCLENCHEMENT est $\geq 1/3\ V_{CC}$, la sortie de ce comparateur reste au niveau BAS et n'a aucun effet sur la bascule. Quand DÉCLENCHEMENT devient $< 1/3\ V_{CC}$, la sortie de ce comparateur passe au niveau HAUT ce qui impose à la bascule $X = 1$ et

$\overline{X} = 0$. Les conditions définies pour la minuterie sont alors: SORTIE est au niveau HAUT et DÉCHARGE est un circuit OUVERT. Notez que l'entrée DÉCLENCHEMENT ne peut servir à mettre à zéro la bascule; elle ne sert qu'à mettre la bascule à 1.

Remarquez qu'il est possible que les sorties des deux comparateurs soient en même temps au niveau HAUT (quand DÉCLENCHEMENT $< 1/3\ V_{CC}$ et SEUIL $> 2/3\ V_{CC}$). Cela a pour effet de mettre au niveau HAUT les deux entrées de la bascule et d'amener X et $\overline{X}$ tous les deux à 1. Cette situation doit être évitée dans les conditions normales.

RÉINITIALISATION Il s'agit ici d'une entrée logique prioritaire qui sert à initialiser n'importe quand la bascule dans l'état $X = 0$ et $\overline{X} = 1$, et cela *indépendamment* des entrées SEUIL et DÉCLENCHEMENT. RÉINITIALISATION est une entrée vraie au niveau BAS qui doit être raccordée à V_{CC} quand elle ne sert pas.

TENSION DE COMMANDE Cette broche permet de commander les tensions de référence appliquées à chaque comparateur. Dans la majeure partie des applications, cette broche est inutilisée; les tensions de référence sont fixées à $2/3\ V_{CC}$ et $1/3\ V_{CC}$ comme nous l'avons déjà dit. Pour changer ces tensions, l'utilisateur applique une tension sur la borne TENSION DE COMMANDE. Par exemple, si TENSION DE COMMANDE est fixée à 4 V, la tension de référence du comparateur 1 sera 4 V, et celle du comparateur 2 sera 2 V. Évidemment, ceci implique que les tensions qui rendent actives les bornes SEUIL et DÉCLENCHEMENT sont différentes.

Quand la borne TENSION DE COMMANDE est inutilisée, on peut la laisser ouverte. Cependant, on la raccorde souvent à la masse au travers d'un condensateur de 0,01 μF qui filtre le bruit de haute fréquence pouvant perturber le fonctionnement des comparateurs.

5-24 MINUTERIE 555 VUE COMME UN MONOSTABLE

La figure 5-60 nous montre la minuterie 555 câblée de manière à fonctionner comme un monostable. Le condensateur de temporisation est raccordé entre DÉCHARGE et la masse en parallèle avec le transistor Q_1 (figure 5-59). Ce condensateur est également relié à la borne SEUIL, de sorte que la tension v_C est appliquée à cette entrée. Le monostable est déclenché quand le front descendant d'une impulsion arrive à sa borne DÉCLENCHEMENT. Ceci a pour effet de générer une impulsion positive sur la borne SORTIE, dont la durée est conditionnée exclusivement par R_T et C_T.

Supposons qu'au début SORTIE = 0, DÉCHARGE = MASSE (la bascule est dans l'état $X = 0/\overline{X} = 1$) et DÉCLENCHEMENT est constante à + 4 V. Comme DÉCHARGE = MASSE, C_T reste à 0 V et il ne peut être chargé au travers de R_T. Donc SEUIL = v_C = 0 V.

Si DÉCLENCHEMENT reçoit momentanément une impulsion dont le niveau est inférieur à $V_{CC}/3$ = 1,67 V, la sortie passe à 5 V et DÉCHARGE = OUVERT (la bascule est maintenant dans l'état $X = 1/\overline{X} = 0$). Maintenant que DÉCHARGE est OUVERT, C_T peut se charger à travers la résistance R_T et s'approcher de la valeur V_{CC} (voir formes d'ondes). Quand v_C atteint $2/3\ V_{CC}$ = 3,33 V, le SEUIL ramène la SORTIE à 0 V et DÉCHARGE = MASSE. Dans ces conditions le condensateur se décharge rapidement pour atteindre presque 0 V. Le circuit est donc revenu dans sa situation initiale et le restera jusqu'à la prochaine impulsion.

La durée t_p de l'impulsion de sortie est déterminée par le temps que prend le condensateur pour se charger de 0 à 3,33 V. On calcule directement ce temps en utilisant l'équation exponentielle pour v_C.

$$v_C = 5 - 5\epsilon^{-t/\tau}$$

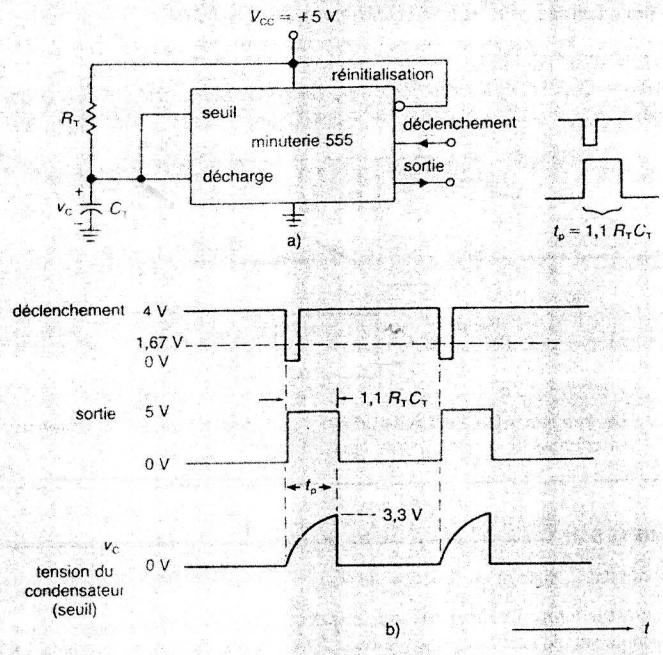


FIGURE 5-60 La minuterie 555 utilisée comme un monostable.

Posons v_C = 3,33 V à $t = t_p$, ce qui donne

$$3{,}33\ \text{V} = 5 - 5\epsilon^{-t_p/\tau}$$

En isolant t_p, on obtient

$$\epsilon^{-t_p/\tau} = 0{,}333$$

$$\frac{-t_p}{\tau} = \ln 0{,}333 = -1{,}1$$

$$t_p = 1{,}1\tau = 1{,}1\ R_T\ C_T$$

Ainsi, chaque impulsion négative appliquée sur DÉCLENCHEMENT donne naissance à une impulsion ayant la durée qu'on vient de calculer. La valeur t_p est indépendante de la durée de l'impulsion de déclenchement *si* l'entrée DÉCLENCHEMENT revient au niveau HAUT avant que C_T ait atteint $2/3\ V_{CC}$. Dans le cas contraire, l'entrée DÉCLENCHEMENT garde la SORTIE au niveau HAUT au-delà du point où SEUIL dépasse 3,3 V. C'est la raison pour laquelle les impulsions appliquées sur DÉCLENCHEMENT sont habituellement beaucoup plus courtes que t_p.

EXEMPLE 5-20

Dites ce qui arrive à la borne SORTIE si une deuxième impulsion DÉCLENCHEMENT arrive pendant l'intervalle t_p.

Solution Rien tant que DÉCLENCHEMENT revient au niveau HAUT avant la fin de l'intervalle t_p (voir figure 5-61). La première impulsion sur DÉCLENCHEMENT met le transistor en conduction ce qui produit la condition: SORTIE = HAUT et DÉCHARGE = OUVERT. La deuxième impulsion n'a pas d'effet puisque le transistor conduit déjà.

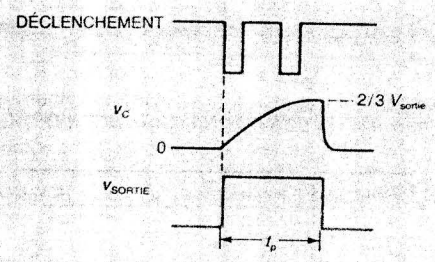


FIGURE 5-61 Les impulsions DÉCLENCHEMENT n'ont aucun effet tant que C_T est en train de se charger (sa tension monte vers 2/3 V_{CC}).

EXEMPLE 5-21

Montrez comment t_p change en fonction de V_{CC}.

Solution t_p est le temps que prend le condensateur pour se charger de 0 à 2/3 V_{CC} quand sa tension monte vers V_{CC}. Il faut donc toujours 1,1 constante de temps pour que C_T atteigne les 2/3 de la valeur de charge maximale, et cela *quelle que soit* cette valeur. *Par conséquent, V_{CC} n'a aucun effet sur t_p.*

Limites relatives à R_T La valeur minimale de R_T est déterminée par le courant maximal qui traverse Q_1 [voir figure 5-62 a)]. Quand DÉCHARGE est une borne à la MASSE, Q_1 conduit et maintient le condensateur à 0 V. Tout le courant débité par V_{CC} et qui traverse R_T est dérivé vers Q_1. Pour le CI 555, la limite de déperdition de courant dans Q_1 est de 10 mA. Donc

$$R_T(\text{MIN}) = \frac{V_{CC}}{10\text{ mA}}$$

Par exemple, à V_{CC} = 5 V, R_T (MIN) = 500 ohms.

La limite maximale pour R_T est conditionnée par les courants qui traversent les bornes SEUIL et DÉCHARGE pendant que C_T se charge [voir figure 5-62 b)]. L'entrée SEUIL tire au maximum 250 nA. La borne DÉCHARGE tire au maximum 100 nA (le courant de fuite de Q_1 quand il est bloqué). La somme de ces deux courants doit être retranchée du courant I_T fourni par V_{CC}, ce qui réduit d'autant le courant de charge du condensateur.

Pour être bien certain qu'il y aura du courant pour charger le condensateur, I_T doit être supérieur à 100 nA + 250 nA = 350 nA. Étant donné que I_T diminue à mesure que s'élève la charge aux bornes du condensateur, I_T est minimal quand la tension de C_T atteint les 2/3 de V_{CC} et que la tension aux bornes de R_T est 1/3 V_{CC}. Donc

$$I_T(\text{MIN}) = \frac{1/3\ V_{CC}}{R_T} > 350\text{ nA}$$

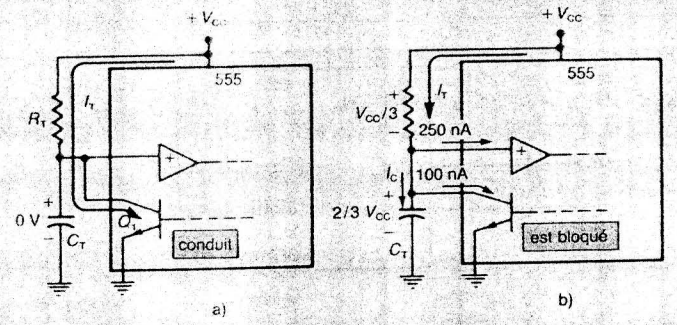


FIGURE 5-62 a) R_T (MIN) est déterminée par le courant de collecteur maximal nominal de Q_1; b) R_T (MAX) est déterminée par le courant de fuite de Q_1 et le courant d'entrée du comparateur.

ou

$$R_T < \frac{V_{CC}}{1{,}050\text{ nA}} \approx \frac{V_{CC}}{1\ \mu\text{A}}$$

ce qui implique

$$R_T(\text{MAX}) \approx \frac{V_{CC}}{1\ \mu\text{A}}$$

Par exemple: pour V_{CC} = 5 V, R_T (MAX) = 5 MΩ et pour V_{CC} = 15 V, R_T (MAX) = 15 MΩ.

EXEMPLE 5-22

On a déjà dit que l'impulsion DÉCLENCHEMENT devait être plus courte que la durée t_p de l'impulsion sur la SORTIE. Alors, dites comment faire pour qu'une impulsion de longue durée déclenche le monostable 555.

Solution Pour pouvoir utiliser une impulsion DÉCLENCHEMENT de longue durée, on doit transformer cette dernière en un décrochement ou une impulsion très étroite. Un circuit *RC* différentiateur (passe-haut) est tout désigné pour cela, comme le montre la figure 5-63. Sur cette figure R_1 et C_1 convertissent l'impulsion d'entrée e_{en} en une pointe dont la durée est inférieure au temps t_p de la SORTIE fixée par R_T et C_T.

Notez que R_1 est raccordé à V_{CC} plutôt qu'à la masse. Ainsi la borne DÉCLENCHEMENT est polarisée normalement à + 10 V, son état inactif. Les fronts de e_{en} produiront des pointes de 10 V au-dessus et au-dessous de cette tension de polarisation de 10 V. La pointe négative place sur DÉCLENCHEMENT une tension inférieure à 1/3 V_{CC}, ce qui déclenche le monostable.

La constante de temps R_1-C_1 est choisie pour produire une pointe négative dont la durée est inférieure à t_p. C'est-à-dire

$$5\ R_1C_1 < t_p = 1{,}1\ R_TC_T$$

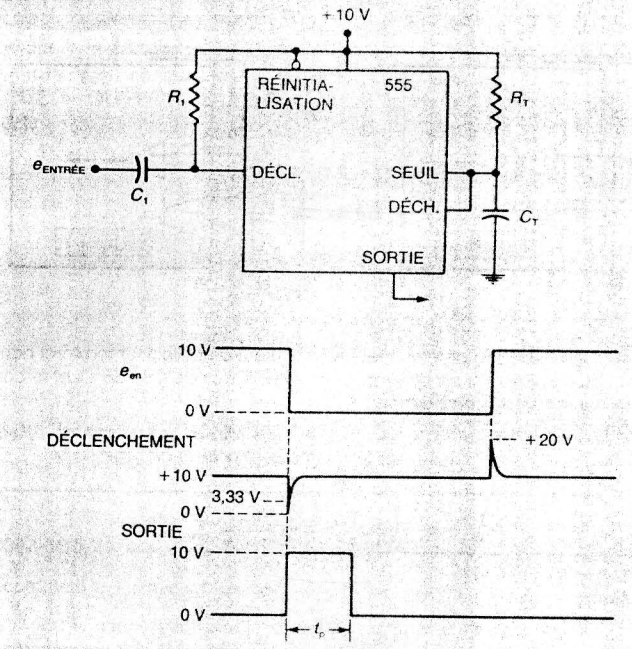


FIGURE 5-63 Le différentiateur (R_1-C_1) convertit la longue impulsion e_{en} en des pointes étroites appliquées à l'entrée DÉCLENCHEMENT.

Ceci nous garantit que DÉCLENCHEMENT sera revenue au-delà de $V_{CC}/3$ avant que prenne fin l'intervalle t_p.

5-25 ANALYSE DE CIRCUITS SÉQUENTIELS

Beaucoup de circuits logiques renferment des bascules, des monostables et des portes logiques interconnectés en vue d'effectuer une opération bien définie. Très fréquemment, un signal d'horloge principal provoque les changements des états dans ce circuit selon une séquence bien précise. La plupart du temps, il est possible d'analyser les circuits séquentiels en respectant la marche à suivre systématique que voici:

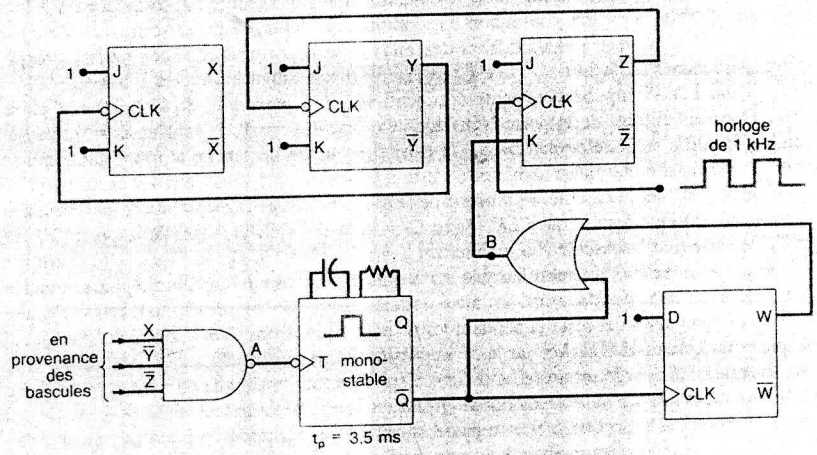
1. Examinez le schéma du circuit et repérez les structures qui vous sont familières comme les compteurs ou les registres à décalage. Ceci peut vous aider à faire l'analyse.
2. Déterminez les niveaux logiques présents sur les bornes d'entrée de chaque bascules *avant* l'arrivée de la première impulsion d'horloge.
3. Utilisez ces niveaux pour trouver quelles seront les sorties des bascules après cette première impulsion.
4. Répétez les étapes 2 et 3 pour les impulsions d'horloge successives.

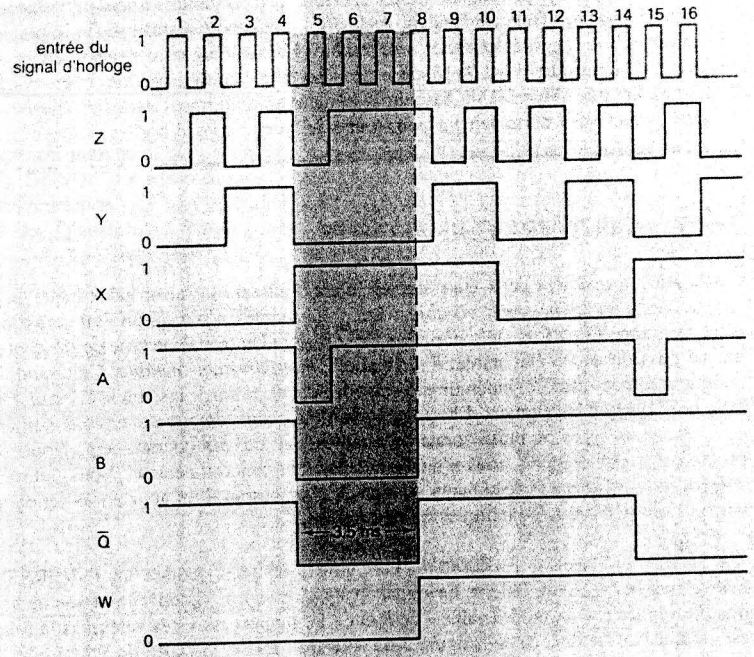
Voici un exemple de cette marche à suivre:

EXEMPLE 5-23

Soit le circuit de la figure 5-64. Initialement, toutes les bascules et le monostable sont à l'état 0 avant qu'arrivent les impulsions d'horloge, dont la fréquence de répétition est 1 kHz. Tracez les formes d'ondes aux points X, Y, Z, W, $\overline{Q}$, A et B pour 16 cycles du signal d'horloge.

FIGURE 5-64 Exemple 5-23.





Solution Initialement, les bascules et le monostable sont à l'état 0, de sorte que $X = Y = Z = W = Q = 0$. Les entrées de la porte NON-ET sont $X = 0$, $\overline{Y} = 1$ et $\overline{Z} = 1$, sa sortie est donc $A = 1$. Les entrées de la porte OU sont $W = 0$ et $\overline{Q} = 1$, sa sortie est donc $B = 1$.

Tant que B reste au niveau HAUT, la bascule Z a ses deux entrées J et K à 1, et cette bascule est donc en mode de basculement. Les bascules X et Y se trouvent aussi dans le même mode puisque leurs entrées J et K sont fixées en permanence à 1. Il est alors évident que les bascules X, Y et Z fonctionnent comme un compteur de 3 bits tant que B reste à 1. Par conséquent, Z commute à chaque front descendant du signal d'horloge, Y commute à chaque front descendant de la sortie Z et X commute à chaque front descendant de la sortie Y. C'est ce qui se passe pendant les cycles 1 à 4.

Quand la 4e impulsion effectue sa transition 1-0, Z passe à 0, Y passe à 0 et X passe à 1. Donc les entrées de la porte NON-ET deviennent $X = 1$, $\overline{Y} = 1$ et $\overline{Z} = 1$, ce qui amène la sortie de A au niveau BAS. Ce front descendant déclenche le monostable qui se retrouve dans son état quasi stable et $\overline{Q}$ reste à 0 pendant 3,5 ms. Ce front descendant de $\overline{Q}$ n'affecte en rien la bascule W, puisque cette bascule de type D réagit aux fronts montants du signal appliqué à sa borne *CLK*. Avec $\overline{Q} = 0$ et $W = 0$, la sortie de la porte OU est à 0, de sorte que la bascule Z a maintenant $J = 1$ et $K = 0$. On sait que cette condition implique la *mise à 1* de cette bascule au moment de l'impulsion d'horloge *suivante*; et cette bascule va rester dans cet état aussi longtemps que $J = 1$ et $K = 0$. Donc Z passe à 1 à la 5e impulsion et reste là pendant les 6e et 7e impulsions (*Nota*: A revient à 1 quand Z passe au niveau HAUT lors de la 5e impulsion puisque $\overline{Z} = 0$).

Après 3,5 ms, le monostable revient à son état stable, c.-à-d. $Q = 0$ et $\overline{Q} = 1$. Le front montant qui survient sur la borne $\overline{Q}$ place la sortie de la bascule W à $D = 1$. Ce qui transmet un 1 à la sortie de B et replace la bascule Z en mode de basculement puisque $J = K = 1$. Le compteur réagit donc comme il se doit aux autres impulsions successives.

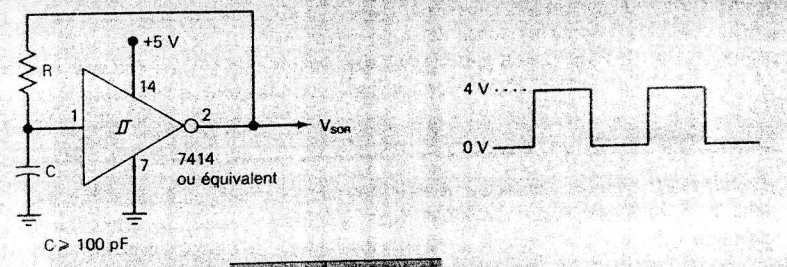
Au moment du front descendant de la 14e impulsion, la sortie A de la porte NON-ET revient à 0, ce qui déclenche le monostable. La sortie de ce dernier n'affecte en rien l'entrée K de Z puisque W garde la sortie de la porte OU au niveau HAUT. En outre, étant donné que l'entrée D est à 1, la bascule W demeure indéfiniment dans l'état $W = 1$, et le compteur fonctionne normalement pour toutes les impulsions d'horloge suivantes.

5-26 MULTIVIBRATEURS ASTABLES

Les bascules possèdent deux états stables, c'est d'ailleurs la raison pour laquelle, on les appelle aussi *multivibrateurs bistables*. Les monostables n'ont qu'un état stable et sont, de ce fait, appelés parfois *multivibrateur monostable*. Un troisième type de multivibrateur, lui, ne possède aucun état stable; il s'appelle le *multivibrateur astable*. Ce type de circuit logique commute (oscille) indéfiniment entre deux états de sortie instables. C'est un montage utile pour obtenir les signaux d'horloge servant au cadencement des circuits numériques.

Plusieurs types de multivibrateurs astables sont utilisés couramment. Nous vous en présentons deux ici, mais nous n'analysons pas leur fonctionnement. Vous pourrez ainsi construire un générateur de signaux si vous en avez besoin d'un pour un projet ou pour la vérification de circuits numériques en laboratoire.

Oscillateur à bascule de Schmitt La figure 5-65 nous montre un INVERSEUR de Schmitt connecté en oscillateur. Le signal en V_{SOR} est approximativement une onde carrée dont la fréquence dépend des valeurs de R et C. Le rapport entre ces valeurs et la fréquence est indiqué dans le tableau de la figure 5-65 pour trois INVERSEURS de Schmitt différents.



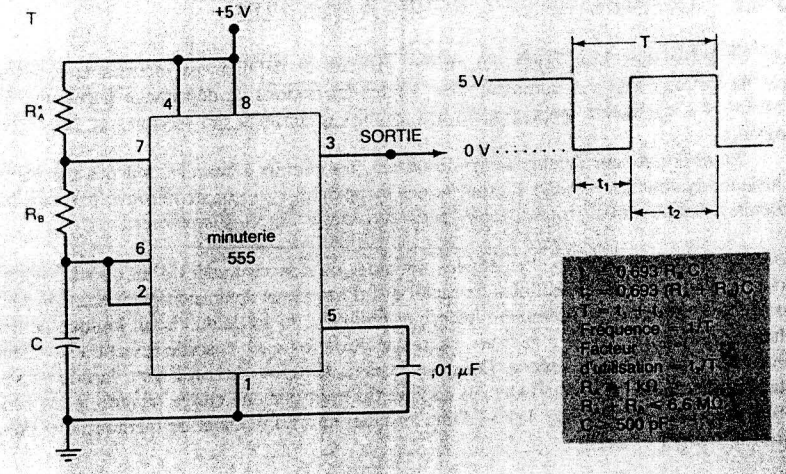
CI	Fréquence	
7414	≈ 0,8/RC	(R ≤ 500 Ω)
74LS14	≈ 0,8/RC	(R ≤ 2 kΩ)
74HC14	≈ 1,2/RC	(R ≤ 10 MΩ)

FIGURE 5-65 Oscillateur à bascule de Schmitt. Une porte NON-ET à bascule de Schmitt 7413 ferait aussi bien l'affaire.

Notez la limite maximale imposée à la résistance pour chacun de ces éléments. Le circuit ne parviendra pas à se mettre en oscillation si R est supérieur à cette limite.

La minuterie 555 câblée en multivibrateur Nous avons déjà mentionné en 5-24 que l'on pouvait utiliser la minuterie 555 pour réaliser un multivibrateur monostable. La figure 5-66 illustre comment câbler les composants externes de la minuterie 555 pour obtenir un oscillateur. Sa sortie est un train d'ondes rectangulaires; elle commute entre deux niveaux logiques en suivant des intervalles qui dépendent, pour chacun des niveaux logiques, des valeurs de R et de C. Les formules applicables pour t_1 et t_2, de même que pour la période totale des oscillations, T, sont données dans la figure. La fréquence des oscillations est toujours, l'inverse de T. Comme ces formules l'indiquent, les intervalles t_1 et t_2 ne peuvent être égaux, sauf si R_A devient nul. Ce qui ne doit jamais arriver sous

FIGURE 5-66 Le CI de la minuterie 555 utilisé comme multivibrateur astable.



peine d'avoir un courant très intense qui traverse le dispositif. Il faut donc conclure qu'il est impossible d'obtenir une onde de sortie parfaitement carrée (facteur d'utilisation de 50 pour cent). Toutefois, il est possible de s'approcher très près du 50 pour cent, en prenant $R_B \gg R_A$ (tout en gardant R_A supérieur à 1 kΩ), ce qui donne $t_1 \approx t_2$.

EXEMPLE 5-24

Calculez la fréquence et le facteur d'utilisation de la sortie du multivibrateur astable 555 pour $C = 0{,}001\ \mu F$, $R_A = 2{,}2\ k\Omega$ et $R_B = 100\ k\Omega$.

Solution

$$t_1 = 0{,}693(100\ k\Omega)(0{,}001\ \mu F) = 69{,}3\ \mu s$$
$$t_2 = 0{,}693(102{,}2\ k\Omega)(0{,}001\ \mu F) = 70{,}7\ \mu s$$
$$T = 69{,}3 + 70{,}7 = 140\ \mu s$$
$$f = 1/140\ \mu s = 7{,}29\ kHz$$
$$\text{cycle d'utilisation} = 70{,}7/140 = 50{,}5\ \%$$

Vous constatez que le cycle d'utilisation est très voisin de 50 % (onde carrée), du fait que R_B est beaucoup plus grand que R_A. On peut tendre encore plus vers 50 % en augmentant d'autant la valeur de R_B par rapport à R_A. Par exemple, vous pouvez démontrer que si l'on ramène R_A à 1 kΩ (sa valeur minimale admissible), les résultats deviennent $f = 7{,}18$ kHz et le cycle d'utilisation passe à 50,3 pour cent.

QUESTIONS DE RÉVISION

1. Déterminez la fréquence approximative d'un oscillateur à bascule de Schmitt basé sur un 74HC14 dont $R = 10\ k\Omega$ et $C = 0{,}005\ \mu F$.
2. Déterminez la fréquence approximative et le cycle d'utilisation de la sortie d'une minuterie 555 pour laquelle $R_A = R_B = 2{,}2\ k\Omega$ et $C = 1000\ pF$.

5-27 DÉPANNAGE DE CIRCUITS À BASCULES

Les CI à bascules sont sujets aux mêmes risques de défaillances internes et externes que les circuits logiques combinatoires. Toutes les notions de dépannage introduites au chapitre 4 s'appliquent intégralement aux circuits renfermant des bascules et des portes logiques.

Étant donné leur faculté de mémorisation, les circuits à bascules touchés par un ou plusieurs dysfonctionnements présentent des symptômes que l'on ne retrouve pas dans les circuits combinatoires. Voici la description de certains de ces symptômes:

Entrées en circuit ouvert Les entrées flottantes ou non raccordées d'un circuit logique sont particulièrement susceptibles de capter des fluctuations de tension parasites, appelées *bruit*. Si le bruit est suffisamment intense et prolongé, la sortie du circuit logique pourra changer d'état en réponse à ce bruit. Dans une porte logique, la sortie revient à sa valeur initiale dès que le bruit s'atténue. Dans une bascule, toutefois, la sortie reste dans le nouvel état du fait de sa mémoire. Donc, la capture du bruit par une entrée ouverte a des conséquences plus néfaste dans le cas d'une bascule ou d'un élément de mémoire que dans celui d'une porte logique.

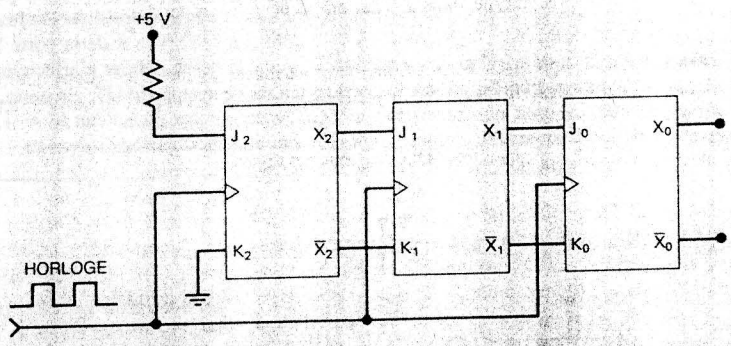
Les entrées de bascule les plus suspectes sont celles qui peuvent déclencher la bascule pour la placer dans un état différent, par exemple *CLK*, *RAU* et *RAZ*. Chaque fois que vous êtes témoin d'une sortie de bascule qui change d'état de manière erratique, envisagez toujours la possibilité qu'une des entrées précédentes soit ouverte.

EXEMPLE 5-25

La figure 5-67 nous montre un registre à décalage de trois bits constitué de bascules TTL. Au départ, toutes les bascules sont dans l'état BAS avant l'application des impulsions d'horloge. Quand les signaux d'horloge sont en œuvre, chaque FM a pour effet de décaler l'information dans chaque bascule vers la bascule de droite. Le schéma illustre la séquence « prévue » des états des bascules après chaque signal d'horloge. Comme $J_2 = 1$ et $K_2 = 0$, la bascule X_2 passe au niveau HAUT à l'instant de l'impulsion d'horloge 1 et y reste pendant les impulsions ultérieures. La valeur HAUTE se propage dans X_1, puis dans X_0 au moment des impulsions d'horloge 2 et 3, respectivement. Ainsi, après la troisième impulsion, toutes les bascules ont la valeur HAUTE en sortie et conservent ces valeurs malgré les impulsions d'horloge qui arrivent sans arrêt.

Maintenant, supposons que la réponse « réelle » des bascules est celle reproduite dans le schéma ci-après. Dans celui-ci, on voit que les bascules ont changé d'état comme prévu pendant les trois premières impulsions. À partir de là, la bascule X_0, plutôt que de rester à 1, comme elle le devrait, alterne entre HAUT et BAS. Trouvez la cause probable de cette défaillance.

FIGURE 5-67 Exemple 5-25.



Numéro de l'impulsion d'horloge	« Prévu » X_2	X_1	X_0	« Réel » X_2	X_1	X_0
0	0	0	0	0	0	0
1	1	0	0	1	0	0
2	1	1	0	1	1	0
3	1	1	1	1	1	1
4	1	1	1	1	1	0
5	1	1	1	1	1	1
6	1	1	1	1	1	0
7	1	1	1	1	1	1
8	1	1	1	1	1	0

Solution À l'instant de la seconde impulsion, X_1 passe au niveau HAUT. Ceci devrait forcer $J_0 = 1$, $K_0 = 0$, et faire en sorte que toutes les impulsions suivantes fixent $X_0 = 1$. Au lieu de cela, nous voyons que X_0 commute à chaque impulsion d'horloge après la seconde impulsion. Ce basculement ne devrait se produire que si J_0 et K_0 était tous les deux à UN. La défaillance plausible est une rupture dans la connexion entre $\overline{X}_1$ et K_0. Rappelons qu'un dispositif TTL réagit à une entrée ouverte comme si la valeur qui s'y trouve était un niveau HAUT, d'où K_0 ouvert est équivalent à un niveau HAUT.

Sorties court-circuitées L'exemple suivant veut montrer comment une défaillance dans un circuit de bascules peut être à l'origine d'un symptôme trompeur et accroître le temps nécessaire pour dépister la faute.

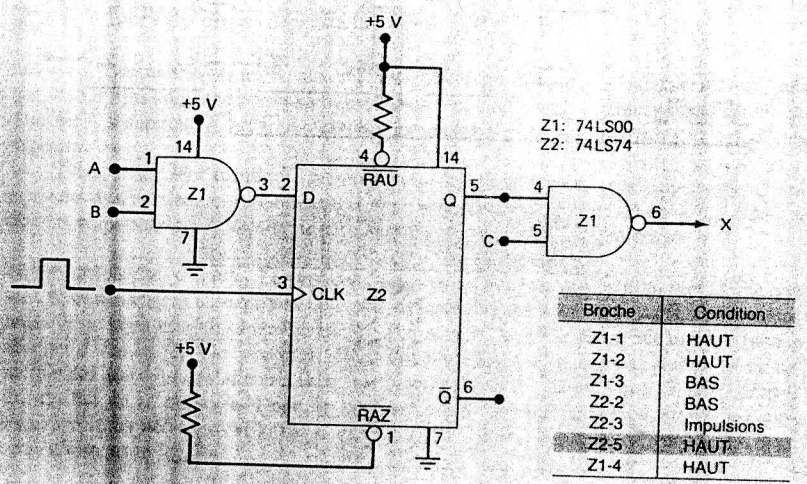
EXEMPLE 5-26

Soit le circuit de la figure 5-68; examinons les indications fournies par la sonde et résumées dans le tableau joint. On voit un BAS sur l'entrée D de la bascule quand les signaux d'horloge sont appliqués à l'entrée *CLK*, alors que la sortie Q ne parvient pas à passer à l'état BAS. Le technicien qui teste le circuit soulève les défaillances possibles suivantes:

1. Z2-5 est court-circuité à V_{CC} par l'intérieur.
2. Z1-4 est court-circuité à V_{CC} par l'intérieur.
3. Z2-5 ou Z1-4 est court-circuité à V_{CC} par l'extérieur.
4. Z2-4 est court-circuité à la MASSE par l'intérieur ou l'extérieur. Cela garde $\overline{RAU}$ activé et force un niveau HAUT malgré les signaux d'horloge de *CLK*.
5. Il y a un bris interne dans Z2 qui empêche Q de répondre correctement à ses entrées.

Le technicien, après ses vérifications ohmmétriques, rejettent les quatre premières possibilités. Il vérifie également les broches Z2, V_{CC} et MASSE et constate que leur tension

FIGURE 5-68 Exemple 5-26.



Broche	Condition
Z1-1	HAUT
Z1-2	HAUT
Z1-3	BAS
Z2-2	BAS
Z2-3	Impulsions
Z2-5	HAUT
Z1-4	HAUT

est correcte. Il est assez réticent à déssouder Z2 du circuit tant qu'il ne sera pas convaincu que c'est la pièce défaillante. Il décide alors d'examiner le signal d'horloge. Il a recours pour cela à un oscilloscope pour vérifier l'amplitude, la fréquence, la durée d'impulsion et les temps de transition. Il constate que toutes ces mesures sont à l'intérieur des valeurs de la fiche technique du 74LS74. Finalement, il conclut que Z2 est la pièce défaillante.

Il retire la puce 74LS74 et la remplace par une autre. Pour sa plus grande déconvenue, il s'aperçoit que le circuit avec la nouvelle puce fonctionne exactement comme avant. Après y avoir bien réfléchi, il décide de changer la puce de la porte NON-ET, sans savoir vraiment pourquoi. Comme prévu, cette manœuvre ne règle rien.

De plus en plus perplexe, notre technicien se souvient du conseil de son professeur de travaux pratiques en électronique: toujours examiner attentivement la plaquette de circuit. Ce qu'il fait. C'est alors qu'il remarque un pont de soudure entre la broche 6 et la broche 7 de Z2. Il élimine ce pont et le circuit se remet à fonctionner correctement.

Expliquez comment ce court-circuit peut donner lieu au fonctionnement observé.

Solution Le pont de soudure court-circuitait la sortie $\overline{Q}$ à la MASSE, la gardant en permanence dans l'état BAS. Rappelons que dans tous les éléments de mémoire et les bascules, les sorties $\overline{Q}$ et Q sont rétrocouplées par l'intérieur, de sorte que le niveau d'une sortie affecte le niveau de l'autre. Par exemple, revenons sur les circuits internes de la bascule J-K de la figure 5-23. Notez comment un niveau BAS constant sur $\overline{Q}$ maintient à un niveau BAS une des entrées de la porte NON-ET, ce qui fait en sorte de garder au niveau HAUT Q quelle que soient les conditions sur J, K et *CLK*.

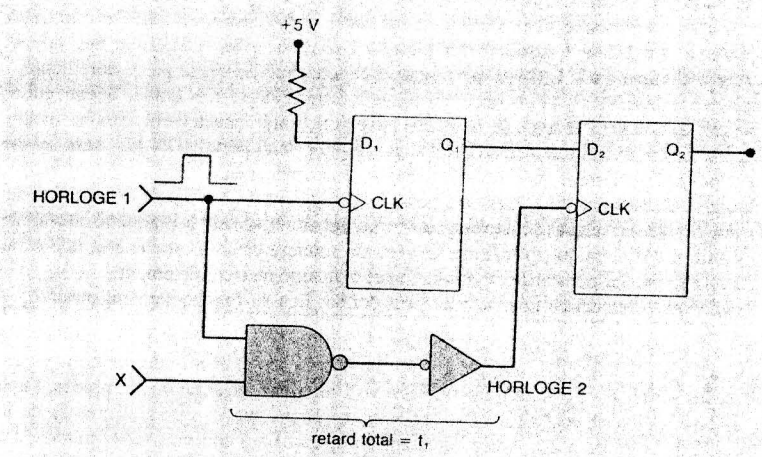
Le technicien en est quitte pour une leçon profitable sur le dépannage des bascules, car il a appris qu'il faudrait toujours vérifier les deux sorties à la recherche d'une défaillance, même celles qui ne sont pas raccordées à d'autres dispositifs.

Dérive d'horloge Un des problèmes de synchronisation les plus courants que l'on rencontre dans les circuits synchronisés est la *dérive d'horloge*. Une certaine dérive d'horloge s'observe quand un signal d'horloge, du fait des retards de propagation, arrive sur les bornes d'entrée *CLK* de différentes bascules à des instants différents. Dans de nombreuses situations, une telle dérive peut mettre une bascule dans un état incorrect. Illustrons cela à l'aide d'un exemple.

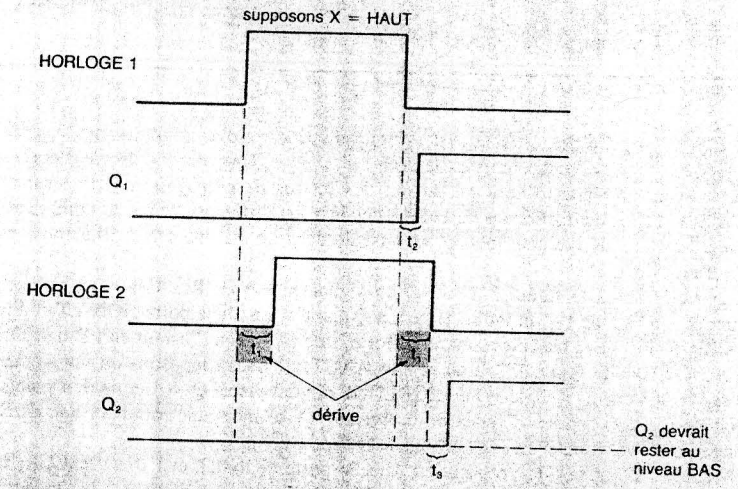
Reportez-vous à la figure 5-69 a) où un signal appelé HORLOGE 1 est connecté directement à la bascule Q_1, et indirectement à Q_2 au travers d'une porte NON-ET et d'un INVERSEUR. Les deux bascules sont, en théorie, déclenchées à l'instant du FD de HORLOGE 1, à la condition que X soit au niveau HAUT. Si nous supposons qu'initialement $Q_1 = Q_2 = 0$ et que $X = 1$, le FD de HORLOGE devrait placer Q_1 à 1 et ne rien changer de l'état de Q_2. Les formes d'ondes de la figure 5-69 b) montrent comment une dérive d'horloge peut occasionner un déclenchement inapproprié de Q_2.

À cause des retards de propagation dans la porte NON-ET et l'INVERSEUR, les fronts du signal HORLOGE 2 sont décalés d'une durée t_1 par rapport à ceux du signal HORLOGE 1. Le FD de HORLOGE 2 arrive sur l'entrée *CLK* de Q_2 en retard de t_1 après l'instant où le FD de HORLOGE 1 est arrivé sur l'entrée *CLK* de Q_1. Cette durée t_1 est appelée dérive de l'horloge. Le FD de HORLOGE 1 commande le passage de Q_1 au niveau HAUT après un temps t_2, le retard de propagation t_{PLH} de Q_1. Si t_2 est inférieur à t_1, Q_1 se trouve déjà au niveau HAUT quand le FD de HORLOGE 2 survient, ce qui attribue incorrectement le niveau un à Q_2, si le temps de stabilisation t_S est écoulé.

Par exemple, supposons que la dérive d'horloge soit 40 ns et que t_{PLH} de Q_1 soit 25 ns. Il s'ensuivra que Q_1 sera au niveau HAUT 15 ns avant le FD de HORLOGE 2. Si le temps de stabilisation de Q_2 est inférieur à 15 ns, Q_2 répond à la présence d'un



a)



t_1 = dérive = somme des retards de la porte NON-ET et de l'INVERSEUR
$t_2 = t_{PLH}$ de Q_1
$t_3 = t_{PLH}$ de Q_2

b)

FIGURE 5-69 Une dérive d'horloge se produit quand deux bascules, supposées se déclencher simultanément, sont commandées à des instants différents, à cause du signal d'horloge de la seconde bascule qui arrive en retard.

niveau HAUT sur son entrée D, à l'instant du FD de HORLOGE 2 et se met lui aussi dans l'état HAUT. Ceci, évidemment, ne correspond pas à la réponse prévue de Q_2, qui devrait rester au niveau BAS.

Les effets de la dérive d'horloge ne sont pas toujours faciles à déceler, étant donné que la bascule touchée réagit de façon inconstante (parfois elle fonctionne correctement, d'autres fois, non). Tout cela, parce que la situation est fonction des retards de propagation des circuits et des paramètres de synchronisation des bascules, paramètres qui varient avec la température, la longueur des chemins de connexion, la tension d'alimentation et la charge. Parfois, le simple fait de raccorder une sonde d'oscilloscope à une bascule ajoute suffisamment de capacité de charge pour augmenter le retard de propagation du circuit et entraîner le fonctionnement correct du circuit; quand la sonde est retirée, le fonctionnement indésirable revient. Voilà le genre de situation qui n'a rien de réjouissant pour un technicien.

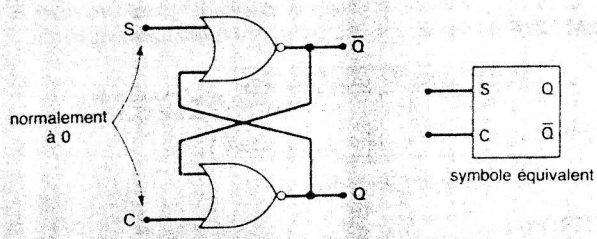
Les problèmes qui ont pour cause une dérive de l'horloge sont réglés par l'égalisation des retards des diverses branches du signal d'horloge, de sorte que le front actif arrive à chaque bascule approximativement au même instant. Cette solution est examinée au problème 5-59.

QUESTION DE RÉVISION

1. Définissez la dérive d'horloge. De quelle manière cause-t-elle des problèmes?

5-28 RÉCAPITULATION DES BASCULES

1. ÉLÉMENT DE MÉMOIRE EN PORTES NI

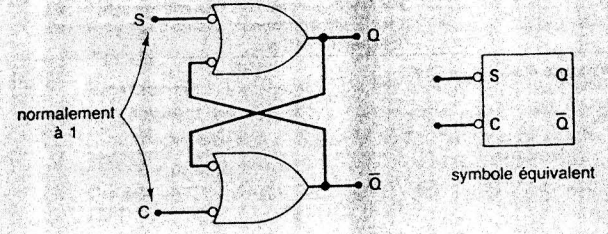


S	C	Q
0	0	inchangée
1	0	Q = 1
0	1	Q = 0
1	1	non valide

FIGURE 5-70

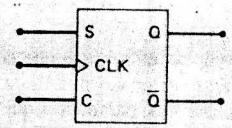
2. ÉLÉMENT DE MÉMOIRE EN PORTES NON-ET

FIGURE 5-71



S	C	Q
0	0	non valide
1	0	Q = 0
0	1	Q = 1
1	1	inchangée

3. BASCULE S-C SYNCHRONE DÉCLENCHÉE PAR UN SIGNAL D'HORLOGE

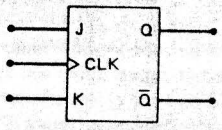


S	C	Q*
0	0	Q_0 inchangée
1	0	1
0	1	0
1	1	ambiguë

* au ↑ de CLK

FIGURE 5-72

4. BASCULE J-K DÉCLENCHÉE PAR UN SIGNAL D'HORLOGE

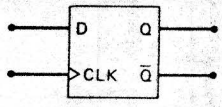


J	K	Q*
0	0	Q_0 inchangée
1	0	1
0	1	0
1	1	$\overline{Q}_0$ bascule

* au ↑ de CLK

FIGURE 5-73

5. BASCULE D DÉCLENCHÉE PAR UN SIGNAL D'HORLOGE

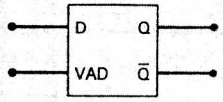


D	Q*
0	0
1	1

* au ↑ de CLK

FIGURE 5-74

6. ÉLÉMENT DE MÉMOIRE D



VAD	D	Q*
0	X	inchangée
1	0	0
1	1	1

Q* suit l'entrée D quand VAD est à 1

FIGURE 5-75

7. ENTRÉES ASYNCHRONES

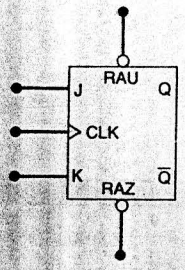


FIGURE 5-76

RAU	RAZ	Q*
1	1	Sans effet; bascule réagit à J, K et CLK
1	0	Q = 0 indépendamment des entrées synchrones
0	1	Q = 1 indépendamment des entrées synchrones
0	0	Ambiguë (ne sert pas)

* N'importe quel état de CLK

PROBLÈMES

SECTIONS 5-1 à 5-3

5-1 Supposez que $Q = 0$ au début, appliquez les formes d'ondes x et y de la figure 5-77 aux entrées S et C d'une mémoire en PORTES NON-ET et déterminez les formes d'onde de Q et $\overline{Q}$.

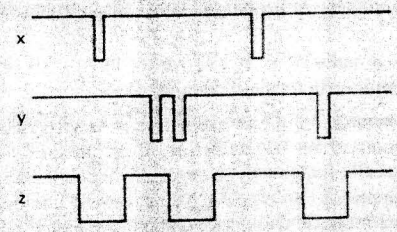


FIGURE 5-77

5-2 Intervertissez les formes d'ondes de la figure 5-77, puis appliquez-les aux entrées S et C d'une mémoire en PORTES NI; déterminez les formes d'ondes de Q et $\overline{Q}$. Supposez que $Q = 0$ au départ.

5-3 Les formes d'ondes de la figure 5-77 sont appliquées au circuit de la figure 5-78. Supposez que $Q = 0$ au départ; déterminez la forme d'onde de Q.

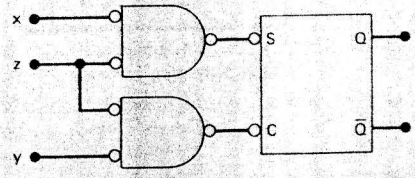


FIGURE 5-78

5-4 Modifiez le circuit de la figure 5-9 afin de pouvoir utiliser une mémoire en PORTES NI.

5-5 Modifiez le circuit de la figure 5-12 afin de pouvoir utiliser une mémoire en PORTES NON-ET.

5-6 Reportez-vous à la figure 5-13. Un technicien teste le fonctionnement de ce circuit en observant ses sorties sur un oscilloscope à mémoire pendant qu'il fait passer l'interrupteur de A à B. Quand l'interrupteur va de A à B, l'écran de l'oscilloscope affiche X_B illustré à la figure 5-79. Quelle défaillance peut donner lieu à un tel comportement? (*Suggestion:* Quelle est la fonction d'un élément de mémoire NON-ET?)

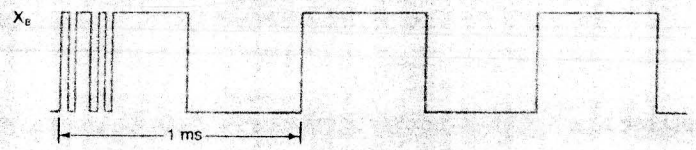


FIGURE 5-79

SECTIONS 5-4 et 5-5

5-7 Une bascule synchrone a $t_S = 20$ ns et $t_M = 5$ ns. Indiquez combien de temps les entrées de commande doivent demeurer stables avant l'arrivée du front de déclenchement.

5-8 Appliquez les formes d'ondes *S*, *C* et *CLK* de la figure 5-17 à la bascule de la figure 5-18, puis déterminez la forme d'onde de *Q*.

5-9 Une bascule de *commutation* n'a qu'une entrée et sa sortie change d'état à chaque fois qu'une impulsion arrive sur son entrée. Il est possible de câbler une bascule S-C synchrone pour qu'elle fonctionne en mode de commutation; voir figure 5-80. La forme d'onde appliquée à l'entrée *CLK* est celle d'une onde carrée de 1 kHz. Assurez-vous que ce montage fonctionne comme une bascule de commutation, puis tracez la forme d'onde de la sortie *Q*. Supposez qu'initialement $Q = 0$.

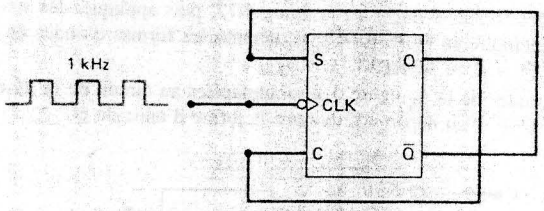


FIGURE 5-80

SECTION 5-6

5-10 Appliquez les formes d'ondes *J*, *K* et *CLK* de la figure 5-21 à la bascule de la figure 5-22. Supposez qu'initialement $Q = 1$ et tracez la forme d'onde de *Q*.

5-11 a) Montrez comment il est possible d'utiliser une bascule J-K comme bascule de commutation. Appliquez une onde carrée de 10 kHz à l'entrée et tracez la forme d'onde de sortie.

b) Connectez la sortie *Q* de la bascule du problème 5-11 a) à l'entrée *CLK* d'une seconde bascule ayant également $J = K = 1$. Trouvez la fréquence de la forme d'onde de sortie de cette seconde bascule.

5-12 Les formes d'ondes illustrées sur la figure 5-81 attaquent deux bascules différentes: a) une bascule J-K déclenchée par un front montant; b) une bascule J-K déclenchée par un front descendant. Dessinez la forme d'onde de la réponse *Q* pour chacune de ces bascules si, au départ, $Q = 0$. Supposez en outre que pour chaque bascule $t_M = 0$.

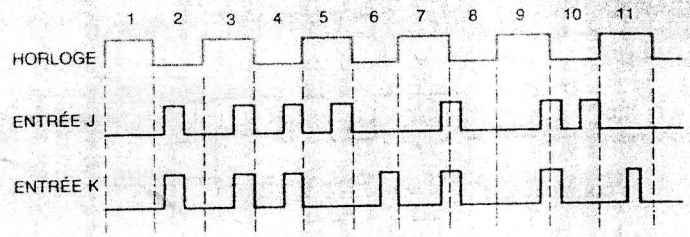


FIGURE 5-81

SECTION 5-7

5-13 Une bascule de type *D* est employée pour *temporiser* une forme d'onde binaire afin que l'information apparaisse sur la sortie un certain temps après son application à l'entrée de la bascule *D*. Déterminez la forme d'onde de *Q* de la figure 5-82, puis comparez-la à la forme d'onde d'entrée. Constatez qu'elle est en retard d'une période du signal d'horloge par rapport à l'entrée. Trouvez une façon d'introduire une temporisation de deux périodes du signal d'horloge.

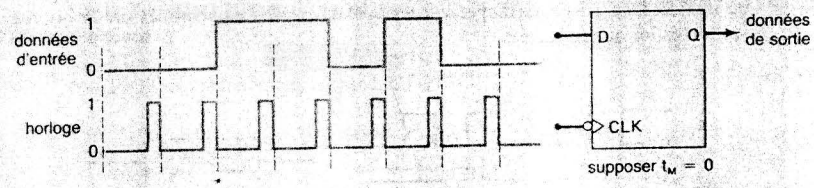


FIGURE 5-82

5-14 Une bascule D déclenchée par un signal d'horloge peut fonctionner en mode de commutation si on la câble comme à la figure 5-83. Supposez qu'initialement $Q = 0$ puis tracez la forme d'onde de *Q*.

5-15 Modifiez le circuit de la figure 5-83 pour que ce soit *Q* qui soit rétrocouplé à *D*. Tracez maintenant la forme d'onde de *Q*.

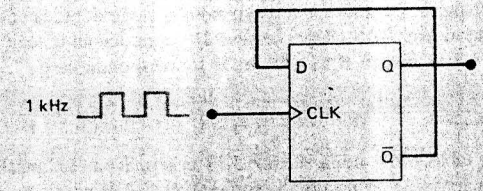


FIGURE 5-83 Bascule D connectée de manière à commuter.

SECTION 5-8

5-16 Comparez le fonctionnement de l'élément de mémoire D avec celui d'une bascule D déclenchée par le front descendant d'un signal d'horloge en leur appliquant

les formes d'ondes de la figure 5-84 puis en traçant les formes d'ondes de leur sortie Q.

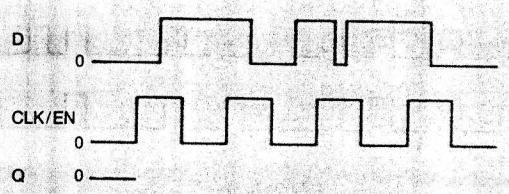


FIGURE 5-84

5-17 Au problème 5-14, nous avons vu qu'il était possible de faire fonctionner en mode de commutation une bascule D déclenchée par un signal d'horloge. Dites pourquoi la même chose est impossible à faire avec un élément de mémoire D.

SECTION 5-9

5-18 Déterminez la forme d'onde de Q pour la bascule de la figure 5-85. Supposez qu'initialement $Q = 0$ et rappelez-vous que les entrées asynchrones ont la priorité sur toutes les autres entrées.

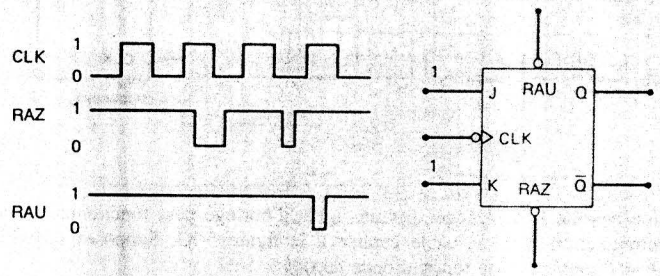


FIGURE 5-85

5-19 Appliquez les formes d'ondes *CLK*, *RAZ* et *RAU* de la figure 5-31 à une bascule D déclenchée par le front montant du signal d'horloge dont les entrées asynchrones sont vraies au niveau BAS. Supposez que D est gardée au niveau HAUT et qu'initialement Q est au niveau BAS. Tracez la forme d'onde de Q.

SECTION 5-11

5-20 Consultez le tableau 5-1 de la section 5-11 et répondez aux questions suivantes:

a) Indiquez le temps qu'il faut à la sortie Q du 4013B pour changer de 0 à 1 après le passage du front actif du signal d'horloge.

b) Dites quelle bascule du tableau 5-1 est celle dont les entrées de commande doivent demeurer stables le plus longtemps après le passage du front actif du signal d'horloge.

c) Indiquez la période de l'impulsion la plus rapide que l'on peut appliquer à la borne *RAU* d'une bascule 7474.

5-21 Reportez-vous au circuit de la figure 5-86. Ce circuit est formé des deux bascules J-K et d'un boîtier 74LS112 agencées de manière que la sortie Q_1 serve de signal *CLK* pour la bascule Q_2. Supposez qu'initialement $Q_1 = Q_2 = 1$, puis trouvez le retard de propagation *total* entre le FD du signal d'horloge et le front descendant de Q_2.

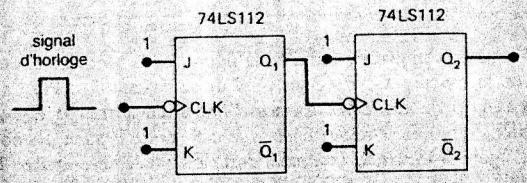


FIGURE 5-86

SECTIONS 5-15 et 5-16

5-22 Modifiez le circuit de la figure 5-40 pour pouvoir utiliser une bascule J-K.

5-23 Dans le circuit de la figure 5-87, les entrées A, B et C sont initialement à 0. La sortie Y ne passe à 1 que lorsque A, B et C passent à 1 en respectant une certaine séquence.

a) Établissez la séquence qui fait passer Y à 1.

b) Expliquez le rôle de l'impulsion DÉBUT.

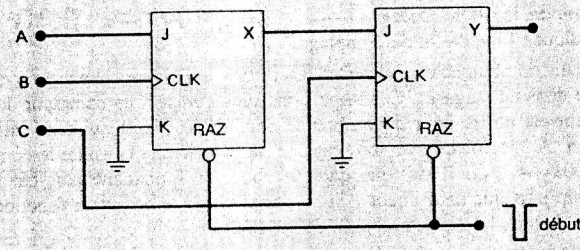


FIGURE 5-87

SECTIONS 5-17 et 5-18

5-24 Dessinez le schéma d'un circuit qui assure le transfert parallèle synchrone des données entre un registre de 3 bits et un autre formé de bascules J-K.

5-25 Refaites le problème 5-24 mais cette fois-ci pour un transfert parallèle asynchrone.

5-26 Un registre à décalage *circulaire* est un registre dans lequel l'information binaire circule tant qu'on applique des impulsions d'horloge. Le registre à décalage de la figure 5-45 peut être converti en un registre circulaire si on connecte X_0 sur la ligne DONNÉES D'ENTRÉE. Aucune autre entrée extérieure n'est nécessaire. Supposez que la valeur installée au début dans le registre circulaire est 1011 (c.-à-d. $X_3 = 1$, $X_2 = 0$, $X_1 = 1$ et $X_0 = 1$). Énumérez la suite des états occupés par les bascules du registre quand on applique huit impulsions de décalage.

5-27 Reportez-vous à la figure 5-46 qui montre un nombre de trois chiffres mémorisé dans le registre X qui est décalé en série dans le registre Y. Indiquez ce qu'il faut faire pour qu'à la fin du transfert, on retrouve le nombre original dans les deux registres. (*Suggestion*: voir problème 5-56.)

SECTION 5-19

5-28 Reportez-vous au compteur binaire de la figure 5-47. Modifiez-le en connectant X_0 sur *CLK* de la bascule X_1 et $\overline{X}_1$ sur *CLK* de la bascule X_2. Placez des 1 dans les trois bascules, puis tracez les formes d'ondes de sortie de X_0, X_1 et X_2 pour 16 impulsions d'entrée. Dressez ensuite le tableau de la suite des nombres, comme celui de la figure 5-48. Un tel compteur porte le nom de *décompteur*. Dites pourquoi.

5-29 Dessinez le graphe des états de ce décompteur et comparez-le au graphe de la figure 5-49. Indiquez leurs différences.

5-30 Montrez comment il est possible d'utiliser des bascules D synchrones dans un compteur comme celui de la figure 5-47. (*Suggestion*: voir problème 5-14.)

5-31 a) Indiquez combien il faut de bascules pour construire un compteur binaire dont l'intervalle de comptage va de 0 à 1023.
b) Calculez la fréquence du signal de sortie de la dernière bascule de ce compteur si la fréquence du signal d'entrée est 2 MHz.
c) Dites quel est le MODULO de ce compteur.
d) Si le compteur est à zéro au début, quel nombre contient-il après 2060 impulsions.

5-32 Un compteur est synchronisé par un signal d'horloge de 256 kHz. La fréquence de sortie de la dernière bascule est 2 kHz.
a) Trouvez le MODULO.
b) Indiquez l'intervalle de comptage.

5-33 Un circuit de photodétecteur sert à lancer une impulsion chaque fois qu'un client pénètre dans un magasin. Ces impulsions synchronisent un compteur de 8 bits. Ce compteur est utilisé pour dénombrer le nombre de clients qui sont entrés dans ce magasin. À la fermeture de son magasin, le propriétaire consulte le compteur et lit $00001001_2 = 9_{10}$. Il sait que quelque chose ne fonctionne pas, car il est entré beaucoup plus que neuf clients. Si on suppose que le compteur fonctionne correctement, donnez la raison probable de cet écart.

5-34 Dessinez le graphe des états d'un compteur MODULO-16.

SECTION 5-20

5-35 Modifiez le circuit de la figure 5-50 pour que la seule adresse qui autorise le transfert de données dans le registre X soit 10110110.

5-36 Supposez que le circuit de la figure 5-50 fonctionne anormalement et que des données sont transférées vers X quand l'adresse est soit 11111110 ou 11111111. Indiquez quelles défaillances des circuits peuvent causer cette anomalie.

SECTION 5-22

5-37 Tracez les formes d'ondes de Q_1, Q_2 et Q_3 qui correspondent à l'impulsion d'entrée de la figure 5-88.

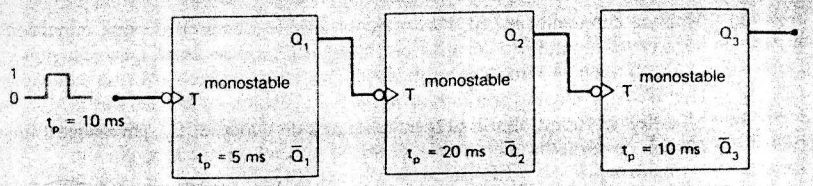


FIGURE 5-88

5-38 Un monostable redéclenchable est employé comme détecteur de fréquence d'une impulsion; il décèle le moment où la fréquence d'une impulsion devient inférieure à une valeur pré-établie. La figure 5-89 montre un exemple simple de cette application. Ce circuit se met en marche quand on ferme momentanément l'interrupteur S_1.
a) Décrivez la réponse du circuit quand les fréquences d'entrée dépassent 1 kHz.
b) Décrivez la réponse du circuit quand les fréquences d'entrée sont inférieures à 1 kHz.
c) Indiquez les modifications qu'il faut apporter à ce circuit pour qu'il décèle les fréquences inférieures à 50 kHz.

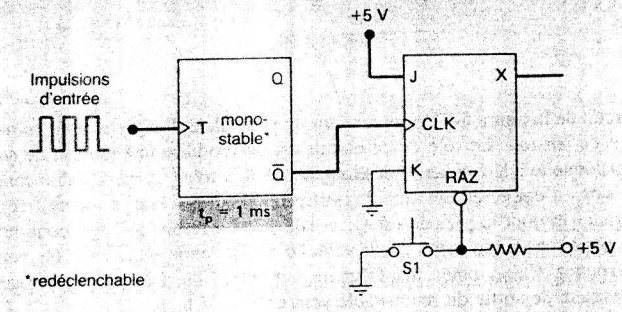


FIGURE 5-89

5-39 Reportez-vous au symbole logique d'un monostable nonredéclenchable 74121 de la figure 5-54 a).
a) Quelles conditions d'entrée sont nécessaires pour que le monostable soit déclenché par un signal sur l'entrée B.
b) Quelles conditions d'entrée sont nécessaires pour que le monostable soit déclenché par un signal sur l'entrée A_1.

5-40 La durée de l'impulsion de sortie d'un monostable 74121 est donnée par la formule approximative:

$$t_p \approx 0{,}7 R_T C_T$$

où R_T est la résistance raccordée entre la broche R_{EXT}/C_{EXT} et V_{CC} et C_T est la capacité entre la broche C_{EXT} et la broche R_{EXT}/C_{EXT}. La valeur de R_T peut varier entre 2 et 40 kΩ et celle de C_T peut aller jusqu'à 1000 μF.

a) Montrez comment on peut raccorder un 74121 pour produire une impulsion négative d'une durée de 5 ms quand l'un ou l'autre de deux signaux logiques (E ou F) tombe à zéro. Normalement, les signaux sur E et sur F sont au niveau HAUT.

b) Modifiez le circuit afin d'intégrer un signal de contrôle, G, qui bloque l'impulsion de sortie du monostable quoiqu'il se produise sur E ou F.

SECTION 5-24

5-41 Soit le circuit de la figure 5-90. Les impulsions d'entrée arrivent au rythme de 100 par seconde. Supposez qu'initialement la bascule A soit à l'état 0 ($A = 0$). Tracez la forme d'onde de sortie.

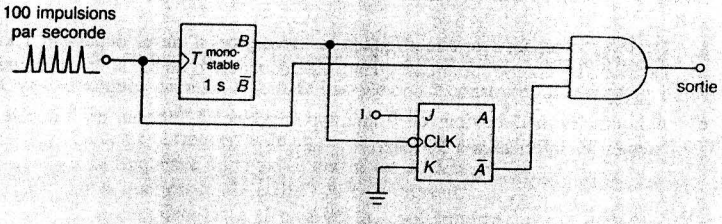


FIGURE 5-90

5-42 Le circuit de la figure 5-91 comprend un monostable et fait partie d'un *convertisseur fréquence-tension*. Le rôle de ce circuit est de produire une tension de sortie c.c. proportionnelle à la fréquence des impulsions d'entrée. Le monostable convertit les impulsions d'entrée en des impulsions rectangulaires ayant la même fréquence de récurrence et dont la période est $t_p = 100\ \mu s$. Le réseau RC agit comme un filtre passe-bas qui ne laisse passer que la valeur c.c. (moyenne) de la sortie du monostable.

a) Trouvez V_{sortie} pour une fréquence d'entrée de 1 MHz. Supposez que les niveaux de sortie du monostable sont entre 0 V et 10 V.

b) Même question que a), mais cette fois-ci pour 2 kHz.

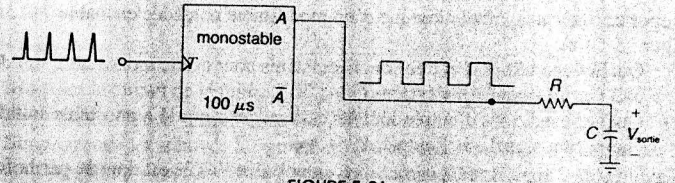


FIGURE 5-91

5-43 Dans le circuit de la figure 5-91, dites quels sont les facteurs qui déterminent les valeurs de R et C du filtre passe-bas. Si la fréquence minimale est 500 Hz, calculez les valeurs appropriées pour R et C. Supposez que la résistance de sortie du monostable est 1 kHz.

5-44 La figure 5-92 montre comment on peut raccorder plusieurs CI 555 pour produire des impulsions de synchronisation séquentielles. Chaque 555 est câblé comme un monostable dont la durée de l'impulsion de sortie est déterminée par R_T et C_T. Le circuit est mis en marche par l'application d'une impulsion DÉBUT à la première minuterie. À la fin de la temporisation, la première minuterie déclenche la deuxième minuterie, qui, à sont tour, déclenche la troisième minuterie. Tracez les formes d'ondes aux bornes SORTIE et DÉCLENCHEMENT de chaque minuterie. Supposez que $R_{entrée}$ à chacune des entrées DÉCLENCHEMENT est très élevée et que R_{sortie} aux bornes SORTIE est très faible.

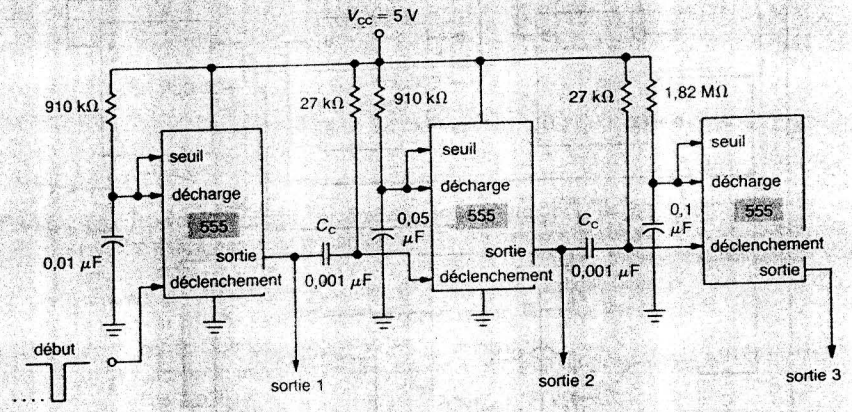


FIGURE 5-92

5-45 Donnez la raison pour laquelle il faut des condensateurs (C_C) dans le circuit de la figure 5-92.

5-46 Démontrez que $t_p = 1.1\ R_T C_T$ pour la minuterie 555, quelle que soit la valeur de V_{CC}.

5-47 Calculez la période t_p maximale que l'on peut obtenir avec le monostable 555, quand sa tension d'alimentation est 10 V et que C_T est 1 μF.

5-48 Reportez-vous au circuit de la figure 5-63. La forme d'onde DÉCLENCHEMENT montre la présence d'une pointe de tension positive qui atteint +20 V. Une tension aussi grande peut endommager le circuit du comparateur 555. Indiquez à quel endroit vous fixeriez une diode *courante* pour supprimer cette pointe. (*Suggestion:* pourquoi ne pas mettre une des bornes de la diode à la masse.)

5-49 L'impulsion $e_{entrée}$ de la figure 5-63 a une durée de 100 ms et ses temps de montée et de descente sont tous les deux de 10 μs. Le monostable utilise $R_T = 4{,}7\ k\Omega$ et $C_T = 0{,}2\ \mu F$. Deux des ensembles suivants de valeurs pour R_1 et C_1 ne produisent *pas* un fonctionnement acceptable. Dites pourquoi.

a) $R_1 = 10\ k\Omega$, $C_1 = 0{,}5\ \mu F$

b) $R_1 = 2{,}2\ k\Omega$, $C_1 = 220\ pF$

c) $R_1 = 10\ k\Omega$, $C_1 = 2000\ pF$

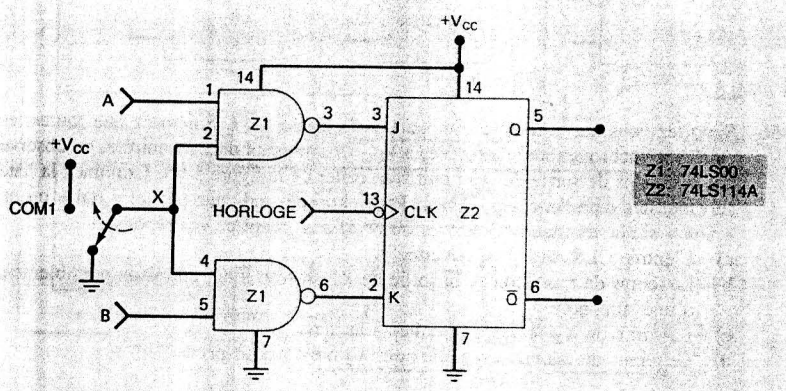
$X_2X_1X_0 = 001$ et $Y_2Y_1Y_0 = 111$. Les impulsions suivantes ne produisent aucun changement de l'état des bascules. Quelles sont les raisons plausibles de ce dysfonctionnement?

5-58 Appliquez à la situation illustrée par la figure 5-69 les deux ensembles de valeurs de synchronisation suivants. Pour chacun, indiquez si la bascule Q_2 fonctionne correctement ou non.

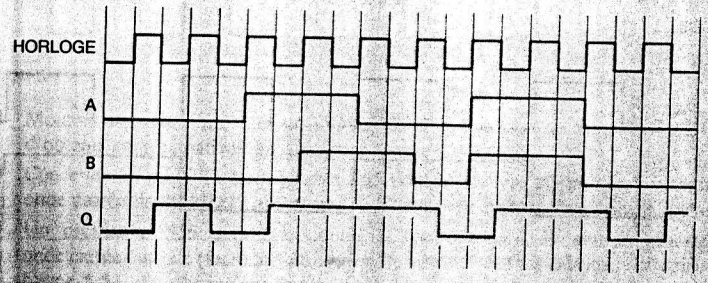
a) *Chaque bascule:* $t_{PLH} = 12$ ns; $t_{PHL} = 8$ ns; $t_S = 5$ ns; $t_M = 0$ ns
La porte NON-ET: $t_{PLH} = 8$ ns; $t_{PHL} = 6$ ns
L'INVERSEUR: $t_{PLH} = 7$ ns; $t_{PHL} = 5$ ns

b) *Chaque bascule:* $t_{PLH} = 10$ ns; $t_{PHL} = 8$ ns; $t_S = 5$ ns; $t_M = 0$ ns
La porte NON-ET: $t_{PLH} = 12$ ns; $t_{PHL} = 10$ ns
L'INVERSEUR: $t_{PLH} = 8$ ns; $t_{PHL} = 6$ ns

5-59 Illustrez comment il est possible d'éliminer le problème de la dérive d'horloge dans la figure 5-69 en insérant aux bons endroits deux INVERSEURS.



a)



b)

FIGURE 5-96

5-60 Reportez-vous à la figure 5-64. Décrivez les modifications de fonctionnement de ce circuit causées par chacune des défaillances suivantes:

a) Entrée du haut de la porte NON-ET court-circuitée à la masse par l'intérieur.

b) Un court-circuit par l'intérieur des entrées de la portes OU.

c) Un circuit ouvert touchant la résistance de synchronisation du monostable.

5-61 En rapport avec la figure 5-96, supposez que les CI sont tous de la famille TTL. La forme d'onde Q est obtenue au moment du test de ce circuit pour les signaux d'entrée illustrés quand le commutateur est à la position « haut ». Ce fonctionnement n'est pas correct. Examinez la liste de défaillances suivantes, et pour chacune, indiquez par un « oui » ou un « non », si elle explique ou non les observations. Justifiez vos réponses.

a) Le point X est toujours au niveau BAS en raison d'un commutateur en mauvais état.

b) La broche 1 de Z1 est court-circuitée à V_{CC} par l'intérieur.

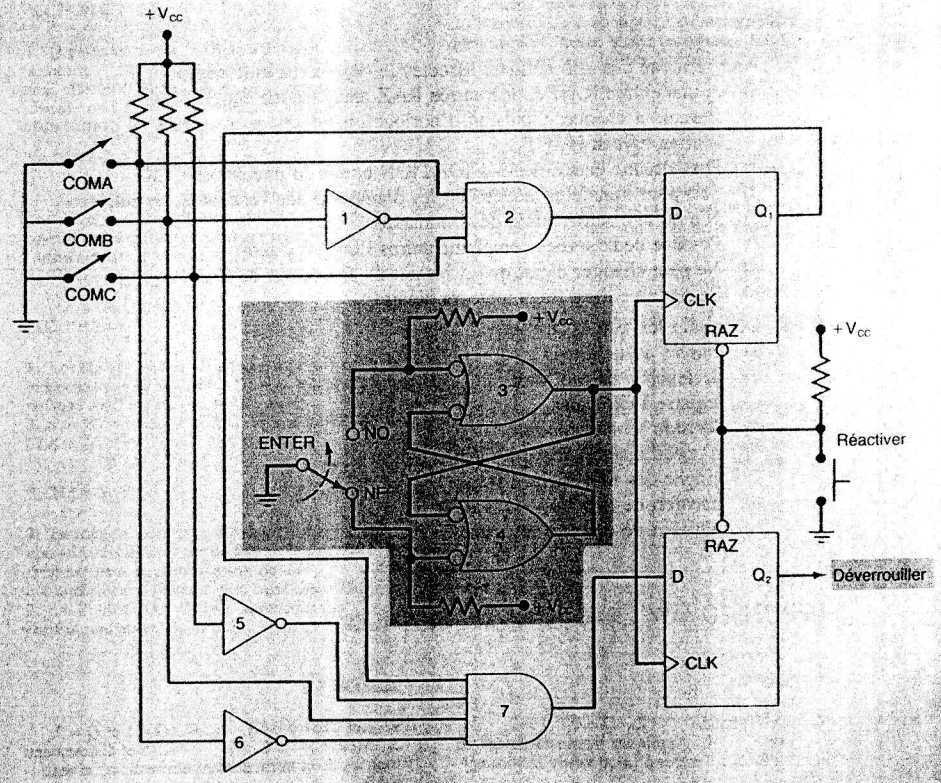
c) La connexion de Z1-3 à Z2-3 est brisée.

d) Il y a un pont de soudure entre les broches 6 et 7 de Z1.

5-62 Le circuit de la figure 5-97 agit exactement comme une serrure à combinaison. Pour faire fonctionner la serrure, procédez comme suit:

1. Pendant un court instant, enfoncez l'interrupteur. Réinitialiser.

FIGURE 5-97



2. Réglez les commutateurs COMA, COMB et COMC pour composer la première partie de la combinaison. Puis, basculez l'interrupteur ENTER d'une position à l'autre.
3. Réglez les commutateurs COMA, COMB et COMC pour composer la seconde partie de la combinaison et basculez ENTER. Cela devrait placer un HAUT sur Q_2 afin de déverrouiller la serrure.

Si une fausse combinaison est composée lors de la première ou de la deuxième étape, la personne doit reprendre du début. Analysez ce circuit et trouvez la combinaison qui déverrouille la serrure.

5-63 Quand la serrure à combinaison de la figure 5-97 est testée, on constate que l'entrée de la bonne combinaison ne déverrouille pas la serrure. Les mesures d'une sonde logique nous apprennent que la composition de la première partie de la combinaison met Q_1 au niveau HAUT, tandis que la composition de la deuxième produit seulement une impulsion sur Q_2. Examinez les problèmes suivants et indiquez lequel ou lesquels peuvent être à l'origine du dysfonctionnement observé. Expliquez vos choix.

a) Les commutateurs COMA, COMB ou COMC rebondissent.
b) L'entrée CLR de Q_2 est ouverte.
c) La connexion entre la sortie de la porte NON-ET 4 et l'entrée de la porte NON-ET 3 est ouverte.

CONTRÔLE

5-64 Pour chacune des affirmations, associez le type de bascule décrit:

a) A une entrée RAU et une entrée RAZ mais aucune entrée CLK.
b) Bascule à chaque impulsion d'horloge quand ses deux entrées de commande sont au niveau HAUT.
c) Possède une entrée de VALIDATION plutôt qu'une entrée CLK.
d) Sert pour transférer facilement des données d'une bascule à une autre.
e) N'a qu'une seule entrée de commande.
f) Possède deux sorties complémentaires l'une de l'autre.
g) Ne peut changer d'état qu'à l'instant du front actif de CLK.
h) Se retrouve dans les compteurs binaires.

5-65 Définissez les termes suivants:

a) entrées asynchrones
b) déclenchée par un front
c) registre à décalage
d) division de la fréquence
e) transfert asynchrone (parallèle)
f) graphe des transitions d'états
g) transfert de données en parallèle
h) transfert de données en série

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 5-1

1. HAUT; haut. 2. $Q = 0$, $\overline{Q} = 1$. 3. Vrai. 4. Appliquer momentanément un niveau BAS à l'entrée S pour mettre la sortie Q à un.

SECTION 5-2

1. Les deux à BAS. 2. $Q = 1$ et $\overline{Q} = 0$. 3. Mettre $S = 0, C = 1$. 4. S et C devraient tous les deux normalement être vrais au niveau BAS.

SECTION 5-4

1. Les entrées de commande synchrones et l'entrée d'horloge. 2. La sortie de la bascule peut changer seulement quand le front d'horloge approprié survient. 3. Vrai. 4. Le temps de stabilisation est le laps de temps spécifié avant le passage du front actif de CLK pendant lequel les entrées de commande doivent rester stables. Le temps de maintien est le laps de temps spécifié après le passage du front actif de CLK pendant lequel les entrées de commande doivent être maintenues stables.

SECTION 5-5

1. Il passe au niveau HAUT. 2. Parce que CLK* est HAUT seulement pendant quelques nanosecondes.

SECTION 5-6

1. Vrai. 2. Non. 3. $J = 1$, $K = 0$.

SECTION 5-7

1. Q passe au niveau BAS au point a et y demeure. 2. Faux. L'entrée D peut changer sans affecter Q, étant donné que Q ne peut changer qu'à l'instant du front actif de CLK.

SECTION 5-8

1. Dans un élément de mémoire D, la sortie Q peut changer quand *VAD* est HAUT. Dans une bascule D, la sortie peut changer seulement à l'instant du front actif de CLK. 2. Faux. 3. Vrai.

SECTION 5-9

1. Les entrées asynchrones fonctionnent indépendamment de l'entrée *CLK*. 2. Oui, puisque *RAU* est vrai au niveau BAS. 3. $J = K = 1$, $RAU = RAZ = 1$ et un FM à *CLK*.

SECTION 5-10

1. Le triangle inclu dans le rectangle désigne une opération déclenchée par un signal; le triangle droit extérieur au rectangle signale un déclenchement à l'instant d'un front descendant. 2. On l'utilise pour désigner la fonction des entrées communes à plus d'un circuit de la puce.

SECTION 5-11

1. t_{PLH} et t_{PHL}. 2. Faux. La forme d'onde doit également satisfaire les exigences $t_W(L)$ et $t_W(H)$.

SECTION 5-17

1. Faux. 2. Bascule D. 3. Six. 4. Vrai.

SECTION 5-18

1. Vrai. 2. Moins d'interconnexions entre les registres. 3. $X_2X_1X_0 = 111$; $Y_2Y_1Y_0 = 101$.

SECTION 5-19

1. 10 kHz. 2. Huit. 3. 256. 4. 2 kHz. 5. $00001000_2 = 8_{10}$.

SECTION 5-21

1. La sortie peut contenir des oscillations. 2. Il donne lieu à des signaux de sortie rapides et bien marqués, même quand les signaux d'entrée passent lentement d'un état à l'autre.

SECTION 5-22

1. $Q = 0$, $\overline{Q} = 1$. 2. Vrai. 3. Les valeurs de R et de C extérieures. 4. Pour un monostable redéclenchable, chaque nouvelle impulsion de déclenchement amorce un nouvel intervalle t_p, quel que soit l'état de la sortie Q.

SECTION 5-26

1. 24 kHz. 2. 218,6 kHz; 66,7 pour cent.

SECTION 5-27

1. La dérive d'horloge c'est l'arrivée d'un signal d'horloge sur les entrées *CLK* de différentes bascules à des instants différents. Elle peut causer un dérèglement dans le fonctionnement d'une bascule.

6

ARITHMÉTIQUE BINAIRE: OPÉRATIONS ET CIRCUITS

Contenu

6-1 Addition binaire
6-2 Écriture des nombres signés
6-3 Addition: complément à 2
6-4 Soustraction: complément à 2
6-5 Multiplication de nombres binaires
6-6 Division binaire
6-7 Addition en DCB
6-8 Arithmétique hexadécimale
6-9 Circuits arithmétiques
6-10 Additionneur binaire parallèle
6-11 Conception d'un additionneur complet
6-12 Additionneur parallèle complet avec ses registres
6-13 Propagation du report
6-14 Additionneur parallèle intégré
6-15 Notation complément à 2
6-16 Additionneur DCB
6-17 Multiplicateurs binaires
6-18 Fonctions de calculs complexes en CI
6-19 Symboles IEEE/ANSI
6-20 Étude d'un cas de dépannage

Objectifs

A la fin de ce chapitre, vous serez en mesure de:

- Effectuer des additions, des soustractions, des multiplications et des divisions sur deux nombres binaires.
- Additionner et soustraire des nombres hexadécimaux.
- Faire la différence entre l'addition binaire et l'addition logique ou booléenne (OU).
- Comparer les avantages et les inconvénients de trois notations différentes utilisées pour représenter les nombres binaires signés.
- Travailler avec des nombres binaires signés exprimés dans la notation complément à 2.
- Comprendre l'additionneur DCB et le processus d'addition de nombres DCB.
- Décrire le fonctionnement de base d'une unité arithmétique et logique.
- Utiliser des additionneurs complets dans la conception d'additionneurs binaires parallèles.
- Indiquer les avantages des additionneurs parallèles munis d'une anticipation du report.
- Analyser le fonctionnement d'un multiplicateur binaire série.
- Décrire certaines des opérations réalisées par une unité arithmétique et logique.
- Reconnaître et interpréter le symbole IEEE/ANSI d'un additionneur parallèle.
- Analyser plusieurs cas de dépannage des additionneurs/soustracteurs.

Introduction

Les diverses opérations arithmétiques qui interviennent dans les ordinateurs et les calculatrices portent sur des nombres exprimés en notation binaire. En tant que telle, l'arithmétique numérique peut être un sujet très complexe, particulièrement si on veut comprendre toutes les méthodes de calcul et la théorie sur laquelle elle s'appuie. Heureusement, il n'est pas nécessaire d'enseigner aux techniciens la théorie complète de l'arithmétique numérique, tout au moins pas avant qu'ils soient devenus des programmeurs expérimentés. Dans ce chapitre, nous allons concentrer nos efforts sur les principes de base qui nous permettent de comprendre comment les machines numériques (c'est à-dire les ordinateurs) réalisent les opérations arithmétiques de base.

D'abord, nous verrons comment on effectue manuellement les opérations arithmétiques en binaire, ensuite nous étudierons les circuits logiques réels qui matérialisent ces opérations dans un système numérique.

6-1 ADDITION BINAIRE

L'addition de deux nombres binaires est parfaitement analogue à l'addition de deux nombres décimaux. En fait, l'addition binaire est plus simple puisqu'il y a moins de cas à apprendre. Revoyons d'abord l'addition décimale:

$$\begin{array}{r} 3\ 7\ 6 \\ +\ 4\ 6\ 1 \\ \hline 8\ 3\ 7 \end{array}$$

chiffre de poids faible

On commence par additionner les chiffres du rang de poids faible, ce qui donne 7. Les chiffres du deuxième rang sont ensuite additionnés, ce qui donne une somme de 13, soit 3 plus un *report* de 1 sur le troisième rang. Pour le troisième rang, la somme des deux chiffres plus le report de 1 donne 8.

Les mêmes règles s'appliquent à l'addition binaire. Cependant, il n'y a que quatre cas qui peuvent survenir lorsqu'on additionne deux chiffres binaires et cela quel que soit le rang. Ces quatre cas sont:

$$0 + 0 = 0$$

$$1 + 0 = 1$$

$$1 + 1 = 10 = 0 + \text{ report de 1 sur le rang de gauche}$$

$$1 + 1 + 1 = 11 = 1 + \text{ report de 1 sur le rang de gauche}$$

Le dernier cas ne se produit que lorsque, pour un certain rang, on additionne deux 1 plus un report de 1 provenant du rang de droite. Voici quelques exemples d'additions de deux nombres binaires:

$$\begin{array}{r} 011\ (3) \\ +\ 110\ (6) \\ \hline 1001\ (9) \end{array} \qquad \begin{array}{r} 1001\ (9) \\ +\ 1111\ (15) \\ \hline 11000\ (24) \end{array} \qquad \begin{array}{r} 11{,}011\ (3{,}375) \\ +\ 10{,}110\ (2{,}750) \\ \hline 110{,}001\ (6{,}125) \end{array}$$

Il n'est pas nécessaire d'étudier des additions ayant plus de deux nombres binaires, parce que dans tous les systèmes numériques les circuits qui additionnent ne traitent pas plus de deux nombres à la fois. Lorsque nous avons plus de deux nombres à additionner, on trouve la somme des deux premiers puis on additionne cette somme au troisième nombre, et ainsi de suite. Ce n'est pas véritablement un inconvénient, puisque les machines numériques modernes peuvent généralement réaliser une opération d'addition en moins d'une microseconde.

L'addition est l'opération arithmétique la plus importante dans les systèmes numériques. Comme nous le verrons, les opérations de soustraction, de multiplication et de division effectuées par les ordinateurs ne sont essentiellement que des variantes de l'opération d'addition.

Comparaison de l'addition binaire et de l'addition logique (OU) Il est très important de saisir la différence entre l'addition logique de l'opération OU et l'addition binaire. L'addition *logique* est une opération booléenne effectuée par une porte OU, qui donne en

sortie 1 quand au moins une de ses entrées est à 1. Par contre, l'addition binaire est une opération *arithmétique* qui produit une somme arithmétique de deux nombres binaires. En conséquence, même si on utilise le même signe + pour l'addition logique et l'addition binaire, il faut être bien conscient que ce signe n'a pas toujours la même signification. Dans ce livre, sa signification est dictée par le contexte. Voici une récapitulation des principales différences entre l'addition binaire et l'addition logique:

addition logique	addition binaire
1 + 1 = 1	1 + 1 = 0 + report de 1
1 + 1 + 1 = 1	1 + 1 + 1 = 1 + report de 1

QUESTION DE RÉVISION

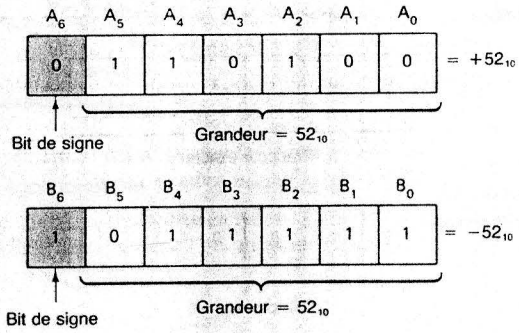
1. Additionnez les paires de nombres binaires que voici: a) 10110 + 00111, b) 011,101 + 010,010, c) 10001111 + 00000001.

6-2 ÉCRITURE DES NOMBRES SIGNÉS

Dans les ordinateurs, on utilise un ensemble d'éléments de mémoire à deux états (habituellement des bascules) pour représenter des nombres binaires. À chaque élément est associé un bit. Par exemple, un registre à bascules de 6 bits peut mémoriser les nombres binaires allant de 000000 à 111111 (0 à 63 en décimal). Ceci représente la *norme ou grandeur* du nombre. Comme la plupart des ordinateurs traitent aussi bien les nombres négatifs que les nombres positifs, il faut adopter une certaine convention pour représenter le *signe* du nombre (+ ou −). Généralement, on ajoute un autre bit au nombre, appelé le **bit de signe**. La convention la plus courante consiste à attribuer au nombre positif le bit de signe 0 et au nombre négatif le bit de signe 1. On peut voir un exemple de ceci à la figure 6.1. Le registre A contient les bits 0110100. Le 0 du bit le plus à gauche (A_6) est le bit de signe qui correspond au signe +. Les autres 6 bits indiquent la grandeur du nombre 110100_2, soit 52 en décimal. Donc le nombre mémorisé dans le registre A est +52. De la même manière, on trouve que le nombre stocké dans le registre B est −52, puisque le bit de signe est 1, ce qui correspond au signe −.

On utilise le bit de signe pour indiquer si le nombre binaire mémorisé est positif ou négatif. Les nombres reproduits à la figure 6-1 sont formés d'un signe de bit et de six bits

FIGURE 6-1 Représentation de nombres binaires signés dans la notation signe-grandeur.



de grandeur. Ces derniers correspondent à l'équivalent binaire exact de la valeur décimale représentée. On parle dans ce cas d'une notation **signe-grandeur** pour représenter des nombres binaires signés.

Bien que la notation signe-grandeur soit directe, les ordinateurs et les calculatrices n'y ont généralement pas recours, en raison de la complexité des circuits qui matérialisent cette notation. On utilise plutôt dans ces machines, pour représenter les nombres binaires signés, la **notation en complément à 2**. Avant d'aborder le déroulement de tout ceci, il importe de voir comment on obtient l'équivalent en complément à 1 et l'équivalent en complément à 2 d'un nombre binaire.

Notation en complément à 1 Le complément à 1 d'un nombre binaire s'obtient en changeant chaque 0 par un 1 et chaque 1 par un 0. Autrement dit, en complémentant chaque bit du nombre. Voici une illustration de cette marche à suivre:

```
1 0 1 1 0 1     nombre binaire initial
↓ ↓ ↓ ↓ ↓ ↓
0 1 0 0 1 0     complément de chaque bit pour obtenir le complément à 1
```

On dit que le complément à 1 de 101101 est 010010.

Notation en complément à 2 Le complément à 2 d'un nombre binaire s'obtient simplement en prenant le complément à 1 de ce nombre et en ajoutant 1 au bit de son rang de poids le plus faible. Voici une illustration de cette conversion pour le cas $101101_2 = 45_{10}$.

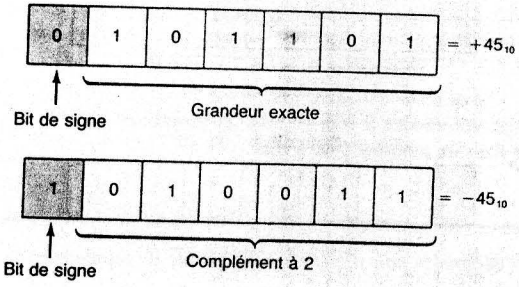
```
  1 0 1 1 0 1     équivalent binaire de 45
  0 1 0 0 1 0     complément de chaque bit pour obtenir le complément à 1
+           1     addition de 1 pour obtenir le complément à 2
  ---------
  0 1 0 0 1 1     le complément à 2 du nombre binaire initial
```

On dit que 010011 est le complément à 2 de 101101.

Écriture de nombres binaires signés dans la notation en complément à 2 Voici comment on écrit des nombres binaires signés en utilisant la notation en complément à 2.

- Si le nombre est positif, sa grandeur est la grandeur binaire exacte et son bit de signe est un 0 devant le bit de poids le plus fort. C'est ce qu'on peut voir sur la figure 6-2 pour $+45_{10}$.

FIGURE 6-2 Écriture de nombres binaires signés dans la notation en complément à 2.



- Si le nombre est négatif, sa grandeur est le complément à 2 de la grandeur exacte et son bit de signe est un 1 à gauche du bit de poids le plus fort. Voyez à la figure 6-2 la représentation du nombre -45_{10}.

La notation en complément à 2 est employée pour exprimer les nombres binaires signés parce que, comme nous le verrons, on parvient grâce à elle à soustraire en effectuant en réalité une addition. Cela n'est pas négligeable dans le cas des ordinateurs, puisqu'avec les mêmes circuits, nous parvenons à soustraire et à additionner.

EXEMPLE 6-1

Écrivez chacun des nombres décimaux signés que voici dans la notation en complément à 2. Utilisez au total 5 bits y compris un signe de bit: a) +13, b) −9, c) +3, d) −2, e) −8.

Solution **a)** Comme c'est un nombre positif, la grandeur (13) est la grandeur exacte, soit $13 = 1101_2$. En ajoutant le signe de bit, on obtient

$$\mathbf{+13 = 01101}$$

bit de signe ↑

b) Étant donné qu'il s'agit d'un nombre négatif, la grandeur (9) est écrite au moyen du complément à 2:

$$9_{10} = 1001_2$$

```
 0110  (complément à 1)
+  1   (ajouter 1 au bit de poids le plus faible)
 ----
 0111  (complément à 2)
```

En ajoutant un bit de signe de 1, le nombre signé complet est

$$\mathbf{-9 = 10111}$$

La marche à suivre précédente comprend deux étapes. D'abord la détermination du complément à 2 de la grandeur, puis l'adjonction d'un bit de signe. Il est possible de tout faire en une étape si on inclut le signe de bit dans la notation en complément à 2. Par exemple, pour trouver la façon d'écrire −9, on utilise le chiffre +9, *y compris le signe de bit*, et on trouve son complément à 2 pour obtenir l'écriture de −9.

$$+9 = 01001$$

```
 10110  (complément de chaque bit)
+  1    (ajouter 1 au bit de poids le plus faible)
 -----
 10111  (complément à 2 de −9)
```

Évidemment nous obtenons le même résultat que précédemment. De façon générale on peut dire que:

la complémentation à 2 d'un nombre signé transforme un nombre positif en un nombre négatif et vice versa.

c) La valeur décimale 3 est représentée en binaire au moyen de 2 bits seulement. Toutefois, l'énoncé du problème demande que l'on utilise une grandeur ayant 4 bits précédée d'un bit de signe. Donc nous avons

$$+3_{10} = 00011$$

Dans de nombreuses situations, le nombre de bits est fixé par la longueur des registres qui contiennent les nombres binaires, d'où la nécessité d'ajouter des 0 pour avoir le nombre de bits requis.

d) Commençons par exprimer +2 au moyen de 5 bits:

+2 =	00010	
	11101	(complément à 1)
+	1	(ajouter 1)
	11110	(complément à 2 du chiffre −2)

e) Commençons par écrire +8:

+8 =	01000	
	10111	(complément à 1)
+	1	(ajouter 1)
	11000	(complément à 2 du chiffre −8)

Conversion des nombres complémentés en binaire La conversion d'un nombre écrit en complément à 1 ou à 2 en sa valeur binaire exacte ne pose pas vraiment de problème. Pour passer du complément à 1 à la valeur binaire exacte, il suffit de complémenter à nouveau chaque bit. De même pour passer du complément à 2 au nombre binaire, il suffit de complémenter chaque bit de la représentation et d'ajouter 1 au bit de poids le plus faible. Dans les deux cas, la conversion en binaire fonctionne de la même façon que la transformation binaire en une forme complémentée. On recommande au lecteur de vérifier ceci en s'exerçant à quelques conversions. Voici un exemple:

Le nombre à convertir	→ **10110**
Le complément à 2	→ 01010
Nouvelle complémentation à 2	→ **10110**

EXEMPLE 6-2

Chacun des nombres suivants est le complément à 2 d'un nombre binaire signé. Trouvez sa valeur décimale: a) 01100, b) 11010, c) 10001.

Solution **a)** Le signe de bit est 0, nous avons donc affaire à un nombre *positif*, les 4 autres bits représentant la grandeur exacte du nombre. Donc $1100_2 = 12_{10}$. Il s'agit donc du nombre décimal **+12**.

b) Le signe de bit étant 1, nous avons donc affaire à un nombre *négatif*, les 4 autres bits sont le complément à 2 de la grandeur. Pour trouver cette grandeur, on doit complémenter de nouveau par rapport à 2. Rappelez-vous que la complémentation à 2 change le signe du nombre signé, de sorte qu'ici on complémente à 2 un nombre binaire négatif (y compris le signe de bit) et on obtient le nombre positif.

11010	(nombre négatif original)
00101	(complément à 1)
+1	(ajouter 1)
00110	

Comme le résultat de la complémentation à 2 est +6, le nombre original doit être −6. Donc

$$11010_2 = \mathbf{-6_{10}}$$

c) En suivant la même procédure qu'en b), il vient

10001	(nombre négatif original)
01110	(complément à 1)
+1	(ajouter 1)
01111	

Donc 10001 = **−15**.

Cas spécial de la notation en complément à 2 Quand un nombre signé a 1 comme bit de signe et que des 0 comme bits de grandeur, son équivalent décimal est -2^N, où N est le nombre de bits de *grandeur*. Par exemple,

$$1000 = -2^3 = -8$$
$$10000 = -2^4 = -16$$
$$100000 = -2^5 = -32$$

et ainsi de suite.

Par conséquent, on peut affirmer que l'intervalle complet des valeurs que l'on peut écrire en complément à 2 au moyen de N bits de grandeur est

$$-2^N \text{ à } +(2^N - 1)$$

Il y a un total de 2^{N+1} valeurs différentes, en *comptant le zéro*.

Par exemple, considérons la liste de tous les nombres signés que l'on peut représenter avec 4 bits au moyen de la notation en complément à 2.

Valeur décimale	Nombre binaire signé dans la notation complément à 2
$-8 = -2^3$	1000
−7	1001
−6	1010
−5	1011
−4	1100
−3	1101
−2	1110
−1	1111
+0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
$7 = 2^3 - 1$	0111

Remarquez que la séquence commence à −8 = 1000 et progresse jusqu'à +7 = 0111, par pas de 0001, comme dans un compteur de 4 bits.

EXEMPLE 6-3

Quel intervalle de valeurs décimales *non signées* est-il possible de représenter avec 8 bits?

Solution En l'absence d'un signe de bit, tous les 8 bits peuvent servir à exprimer la grandeur. Par conséquent, cet intervalle va de:

$$00000000_2 = 0_{10}$$

à

$$11111111_2 = 255_{10}$$

Soit un total de 256 valeurs.

EXEMPLE 6-4

Quel intervalle des valeurs décimales *signées* est-il possible de représenter avec 8 bits?

Solution La plus grande valeur négative que l'on peut écrire est:

$$10000000_2 = -2^7 = -128_{10}$$

La plus grande valeur positive que l'on peut écrire est:

$$01111111_2 = +2^7 - 1 = +127_{10}$$

L'intervalle recherché est donc -128 à $+127$, soit un nombre total de 256 valeurs, si on compte le zéro.

EXEMPLE 6-5

Un ordinateur donné mémorise les deux nombres signés suivants en utilisant la notation en complément à 2. Durant l'exécution d'un programme, il reçoit l'ordre de changer le signe de chacun de ces nombres: soit remplacer $+31$ par -31 et -12 par $+12$. Comment réalisera-t-il cette tâche?

$$00011111_2 = +31_{10}$$

$$11110100_2 = -12_{10}$$

Solution Un nombre signé peut être changé de signe simplement en recomplémentant à 2 *la totalité* du nombre, c'est-à-dire en prenant en compte le bit de signe. Les circuits de l'ordinateur retirent les nombres de la mémoire, en calcule les compléments à 2 et retournent le résultat en mémoire.

QUESTIONS DE RÉVISION

1. Écrivez chacune des valeurs suivantes comme un nombre signé de 5 bits dans la notation en complément à 2: a) $+13$, b) -7, c) -16.
2. Chacun des nombres binaires signés que voici est exprimé dans la notation en complément à 2. Déterminez leur équivalent décimale: a) 100011, b) 1000000, c) 0111111.
3. Dites quel est l'intervalle des valeurs décimales qu'on peut représenter avec 12 bits (y compris le bit de signe).
4. Combien faut-il de bits pour représenter toutes les valeurs décimales comprises entre -50 et $+50$?
5. Quelle est la plus grande valeur décimale négative que l'on peut représenter avec un total de 16 bits.
6. Effectuez la complémentation à 2 de chacun des nombres suivants: a) 10000, b) 10000000, c) 1000.

6-3 ADDITION: COMPLÉMENT À 2

La notation en complément à 2 et la notation en complément à 1 sont très semblables. Toutefois, la notation en complément à 2 jouit généralement de certains avantages quand vient le temps de construire des circuits. Nous allons maintenant étudier comment les machines numériques additionnent et soustraient quand les nombres négatifs sont écrits dans la notation en complément à 2. Dans tous les cas étudiés, il est important que vous remarquiez que le bit de signe de chaque nombre est traité sur le même pied que les bits de la partie grandeur.

Cas I: deux nombres positifs. L'addition de deux nombres positifs est immédiate. Soit l'addition de $+9$ et $+4$:

```
+9 ⟶  0 | 1001    (cumulande)
+4 ⟶  0 | 0100    (cumulateur)
      0 | 1101    (somme = +13)
      ↑
      └ bits de signe
```

Remarquez que les bits de signe du **cumulande** et du **cumulateur** sont 0 et que celui de la somme est aussi 0, ce qui indique un nombre positif. Notez aussi qu'on a fait en sorte que le cumulande et le cumulateur aient le même nombre de bits. Il faut *toujours* s'assurer de cela dans la notation en complément à 2.

Cas II: nombre positif et nombre négatif plus petit. Soit l'addition de $+9$ et de -4. Rappelez-vous que -4 est exprimé dans la notation en complément à 2. Donc $+4(00100)$ doit être converti en $-4(11100)$

```
          bits de signe
          ↓
+9 ⟶  0 | 1001    (cumulande)
−4 ⟶  1 | 1100    (cumulateur)
   X  0 | 0101
   ↑
   └ ce report n'est pas pris en considération de sorte que le résultat
     est 00101 (somme = +5)
```

Dans ce cas-ci, le bit de signe du cumulateur est 1. Remarquez que les bits de signe sont aussi additionnés. En fait, un report est produit au moment de l'addition du dernier rang. *Ce report est toujours rejeté* d'où la somme finale de 00101, soit le nombre décimal $+5$.

Cas III: nombre positif et nombre négatif plus grand. Soit l'addition de -9 et de $+4$:

```
−9 ⟶ 10111
+4 ⟶ 00100    (somme = −5)
     11011
     ↑
     └ bit de signe négatif
```

Dans ce cas-ci le bit de signe de la somme est 1, ce qui indique un nombre négatif. Comme la somme est un nombre négatif, la réponse est le complément à 2 de la grandeur exacte. Donc 1011 est en réalité le complément à 2 de la somme. Pour trouver la grandeur exacte

de la somme, on doit prendre le complément à 2 de 1011, ce qui donne 0101 (5); la réponse est donc 11011 = −5.

Cas IV: deux nombres négatifs

$$\begin{array}{r} -9 \longrightarrow 10111 \\ -4 \longrightarrow 11100 \\ \hline \not{1}10011 \end{array}$$

↑↑ bit de signe

↑ ce report n'est pas pris en considération, le résultat est donc 10011 (somme = −13)

Le résultat définitif est de nouveau négatif et la réponse trouvée est donc le complément à 2 de la réponse précédée du bit de signe 1. Remarquez que 0011 est le complément à 2 de 1101 (13).

Cas V: nombres égaux et opposés

$$\begin{array}{r} -9 \longrightarrow 10111 \\ +9 \longrightarrow 01001 \\ \hline 0 \quad \not{1}00000 \end{array}$$

↑ rejeté, le résultat est donc 00000 (somme = +0)

Le résultat est évidemment +0, comme on s'y attendait.

QUESTIONS DE RÉVISION

1. Vrai ou faux: Quand la somme de deux nombres binaires signés a comme bit de signe 1, la grandeur de la somme est exprimée dans la notation complément à 2.
2. Additionnez les paires suivantes de nombres signés. Exprimez la somme sous forme d'un nombre binaire signé et d'un nombre décimal: a) 100111 + 111011, b) 100111 + 011001

6-4 SOUSTRACTION: COMPLÉMENT À 2

Une opération de soustraction qui porte sur des nombres exprimés dans la notation en complément à 2 est en réalité une opération d'addition qui diffère peu des cas examinés à la section 6-3. Quand on soustrait un nombre binaire (le **diminuteur**) d'un autre nombre binaire (le **diminuande**), la marche à suivre est comme suit:

1. Prendre le complément à 2 du diminuteur, *y compris* son bit de signe. Si ce dernier est un nombre positif, il deviendra un nombre négatif dans la notation en complément à 2. Si le diminuteur est un nombre négatif, la complémentation à 2 en fera un nombre positif écrit en grandeur exacte. Autrement dit, nous changeons le signe du diminuteur.
2. Après avoir complémenté à 2 le diminuteur, on *l'additionne* au diminuande. Le diminuande conserve sa forme initiale. Le résultat de cette addition représente la *différence* recherchée. Le bit de signe de la différence indique si la réponse est positive ou négative et si on est en notation binaire exacte ou en notation en complément à 2.
3. Rappelez-vous que les deux nombres doivent avoir le même nombre de bits.

Examinons la soustraction suivante: +9 − (+4).

diminuande (+9) ⟶ 01001

diminuteur (+4) ⟶ 00100

Changez le diminuteur pour sa version en complément à 2 (11100), ce qui représente −4. Maintenant additionnez-le au diminuande.

$$\begin{array}{rll} & 01001 & (+9) \\ + & 11100 & (-4) \\ \hline \not{1} & 00101 & (+5) \end{array}$$

↑ la retenue est rejetée, le résultat est donc 00101 = +5

Quand on complémente à 2 le diminuteur, on obtient en réalité −4, de sorte qu'on *additionne* +9 à −4, ce qui est équivalent à soustraire de +9 le nombre +4. C'est la même situation que dans le cas II de la section 6-3. En définitive, toute opération de soustraction se résume à une addition lorsqu'on utilise la notation en complément à 2. Cette caractéristique de la notation en complément à 2 explique pourquoi c'est la méthode la plus utilisée, puisqu'on peut additionner et soustraire en utilisant les mêmes circuits.

On recommande au lecteur de vérifier la marche à suivre précédente avec les soustractions que voici: a) +9 − (−4); b) −9 − (+4); c) −9 − (−4); d) +4 − (−4). Rappelez-vous que lorsque le bit de signe de la réponse est 1, vous avez affaire à un nombre négatif exprimé dans la notation en complément à 2.

Dépassement Dans chacun des exemples d'addition et de soustraction que l'on vient d'étudier, les nombres que l'on a additionnés étaient constitués à la fois d'un bit de signe et de 4 bits de grandeur. Les réponses aussi comportaient un bit de signe et 4 bits de grandeur. Tout report fait sur le bit de sixième rang était rejeté. Dans tous les cas étudiés, la grandeur de la réponse ne dépassait jamais la capacité des 4 bits. Voyons l'addition de +9 à +8.

$$\begin{array}{rl} +9 \longrightarrow & 0\ 1001 \\ +8 \longrightarrow & 0\ 1000 \\ \hline & 1\ 0001 \end{array}$$

↑ bit de signe

Le bit de signe de la réponse est celui d'un nombre négatif, ce qui est manifestement une erreur. La réponse devrait être +17. Étant donné que la grandeur est 17, il faut plus de 4 bits pour l'exprimer, et il y a donc un *dépassement* sur le rang du bit de signe. Une condition de dépassement donne toujours lieu à un résultat inexact, et on la détecte en examinant le bit de signe du résultat et en le comparant aux bits de signe des nombres additionnés. Dans un ordinateur, il existe un circuit spécialement conçu pour détecter les conditions de débordement et pour indiquer que la réponse est fausse. Nous étudions un tel circuit dans un des problèmes de la fin du chapitre.

QUESTIONS DE RÉVISION

1. Soustrayez les paires de nombres signés suivants en utilisant la notation en complément à 2. Exprimez les résultats sous forme de nombres binaires signés et sous forme de nombres décimaux: a) 01001 − 11010, b) 10010 − 10011.
2. Indiquez comment on décèle un dépassement arithmétique au moment de l'addition de nombres signés.

6-5 MULTIPLICATION DE NOMBRES BINAIRES

On multiplie les nombres binaires de la même façon qu'on multiplie les nombres décimaux. En réalité, le processus est plus simple car les chiffres du multiplicateur sont toujours 0 ou 1, de sorte qu'on multiplie toujours par 0 ou par 1. Voici un exemple de multiplication de nombres binaires non signés.

```
  1001  <-- multiplicande = 9(10)
  1011  <-- multiplicateur = 11(10)
 ------
  1001
  1001
  0000
 1001
 -------
1100011}   produit final = 99(10)
```

Dans cet exemple, le multiplicande et le multiplicateur sont en notation binaire exacte et il n'y a pas de bit de signe. La marche à suivre est exactement la même que pour les multiplications décimales. D'abord, examinons le bit de poids le plus faible du multiplicateur; dans notre exemple il s'agit d'un 1. Ce 1 multiplie le multiplicande pour donner 1001, c'est notre premier produit partiel. Ensuite, examinons le deuxième bit du multiplicateur. Il s'agit d'un 1, ce qui donne le second produit partiel. Notez qu'en écrivant le second produit partiel on le *décale* d'un rang vers la gauche par rapport au premier. Le troisième bit du multiplicateur est 0 et notre troisième produit partiel est 0000; ce troisième produit est aussi décalé d'un rang vers la gauche par rapport au produit partiel précédent. Le quatrième bit du multiplicateur est 1, de sorte que le dernier produit partiel est 1001 que l'on écrit à nouveau décalé d'un rang vers la gauche. On additionne ensuite les quatre produits partiels pour obtenir le produit final.

La plupart des machines numériques ne peuvent additionner que deux nombres binaires à la fois. C'est la raison pour laquelle les produits partiels d'une multiplication ne peuvent être additionnés ensemble en une seule fois. Ils sont plutôt additionnés deux par deux; c'est-à-dire que le premier est additionné au second, que leur somme est additionnée au troisième et ainsi de suite. C'est ce qu'on illustre ci-après:

```
addition  { 1001  <-- premier produit partiel
          { 1001  <-- second produit partiel décalé vers la gauche
          -------
addition  { 11011 <-- somme des deux premiers produits partiels
          { 0000  <-- troisième produit partiel décalé vers la gauche
          -------
addition  { 011011 <-- somme des trois premiers produits partiels
          { 1001   <-- quatrième produit partiel décalé vers la gauche
          --------
  1100011 <-- somme des quatre produits partiels
              qui égale le produit final
```

Multiplication: complément à 2 Dans les machines qui utilisent la notation en complément à 2, la multiplication est effectuée de la façon décrite ci-dessus quand le multiplicande et le multiplicateur sont exprimés en notation binaire exacte. Si les deux nombres à multiplier sont positifs, ils sont déjà dans cette notation et ils sont multipliés tels quels. Le produit résultant est évidemment positif et son bit de signe est 0. Quand les deux nombres sont négatifs, ils sont donc écrits dans la notation en complément à 2. Chacun de ces nombres est complémenté à 2 pour obtenir un nombre positif et ce sont les résultats de ces complémentations qu'on multiplie. Le produit est un nombre positif dont le bit de signe est 0.

Quand un des nombres est positif et que l'autre est négatif, le nombre négatif est d'abord complémenté à 2 pour obtenir une grandeur positive. Le produit est exprimé selon la notation en grandeur exacte. Cependant, le produit doit être négatif car les nombres à multiplier sont de signes opposés. Par conséquent, on complémente à 2 le produit et on ajoute le bit de signe 1.

QUESTION DE RÉVISION

1. Multipliez 0111 et 1110.

6-6 DIVISION BINAIRE

La division d'un nombre binaire (le *dividende*) par un autre (le *diviseur*) est identique à la division de deux nombres décimaux. En réalité, la division en binaire est plus simple puisque pour déterminer combien de fois le diviseur entre dans le dividende, il n'y a que 2 possibilités 0 ou 1. Voici des exemples de divisions:

```
      0011
  11 | 1001      (9/3 = 3)
       011
       ----
       0011

      0010,1
 100 | 1010,0    (10/4 = 2,5)
       100
       ----
        100
        100
        ---
          0
```

Dans la plupart des ordinateurs modernes, les soustractions qui ont lieu durant une opération de division sont généralement des soustractions avec complément à 2, c'est-à-dire on complémente à 2 le soustracteur puis on effectue l'addition.

La division de nombres signés s'effectue de la même façon que la multiplication. Les nombres négatifs sont complémentés pour obtenir des nombres positifs puis la division est effectuée. Si le dividende et le diviseur sont de signes opposés, le quotient est complémenté à 2 pour obtenir un nombre négatif, puis on lui ajoute un bit de signe de 1. Si le dividende et le diviseur ont le même signe, le quotient est laissé sous sa forme positive et on lui ajoute un bit de signe de 0.

6-7 ADDITION EN DCB

Au chapitre 2, nous avons dit que de nombreux ordinateurs représentent les nombres décimaux au moyen du code DCB. Rappelons que ce code fait correspondre à *chaque* chiffre décimal un code de 4 bits compris entre 0000 et 1001. L'addition de nombres décimaux exprimés sous forme DCB se comprend mieux en étudiant deux cas qui peuvent survenir quand on additionne deux chiffres décimaux.

Somme égale ou inférieure à 9 Additionnons 5 à 4 en utilisant pour chacun leur représentation DCB:

```
  5      0101 ←— DCB de 5
+ 4    + 0100 ←— DCB de 4
  9      1001 ←— DCB de 9
```

L'addition est effectuée comme une addition binaire normale et la somme est 1001, soit le code DCB de 9. Voici un autre exemple: additionnons 45 à 33:

```
 45      0100  0101 ←— DCB de 45
+33    + 0011  0011 ←— DCB de 33
 78      0111  1000 ←— DCB de 78
```

Dans cet exemple, les codes de 4 bits associés à 5 et à 3 sont additionnés selon les règles binaires pour donner 1000, ce qui est le code DCB de 8. De même, l'addition des chiffres décimaux de second rang donne en binaire 0111, ce qui est le code DCB de 7. Le total est 0111 1000, soit le code DCB de 78.

Dans les exemples précédents, aucune somme de deux chiffres décimaux ne dépassait 9; donc, *il n'y a pas eu de reports décimaux*. Dans cette situation l'addition DCB est un processus direct équivalant à l'addition binaire.

Somme supérieure à 9 Additionnons 6 et 7 en DCB:

```
  6      0110 ←— DCB de 6
+ 7    + 0111 ←— DCB de 7
+13      1101 ←— code invalide en DCB
```

La somme 1101 n'existe pas dans le code DCB; il s'agit de l'une des six représentations codées de 4 bits interdites ou non valides. Cette représentation est apparue parce qu'on a additionné deux chiffres dont la somme dépasse 9. Dans un tel cas, il faut corriger la somme en additionnant 6 (0110) afin de prendre en considération le fait qu'on saute six représentations codées non valides:

```
  6        0110 ←— DCB de 6
+ 7      + 0111 ←— DCB de 7
 13        1101 ←— somme non valide
       1  0110 ←— additionner 6 pour corriger
      0001  0011 ←— DCB de 13
        1     3
```

Comme on le montre ci-dessus, l'addition de 0110 à la somme non valide donne la représentation DCB exacte. Notez qu'un report a lieu sur le chiffre décimal de deuxième rang. Il est obligatoire d'additionner 0110 quand la somme de deux chiffres décimaux dépasse 9.

Voyons un autre exemple: additionnez 47 à 35 en DCB:

```
 47      0100  0111 ←— DCB de 47
+35    + 0011  0101 ←— DCB de 35
 82      0111  1100 ←— somme non valide dans le premier chiffre
        1  0110 ←— addition de 6
      1000  0010 ←— somme DCB exacte
        8     2
```


L'addition des codes de 4 bits pour 7 et 5 produit une somme non valide que l'on corrige en additionnant 0110. Notez que ceci produit un report de 1, qui est ajouté à la somme DCB des chiffres du second rang.

Soit l'addition de 59 à 38 en DCB:

```
                1
                ↓
 59      0101  1001 ←— DCB de 59
+38    + 0011  1000 ←— DCB de 38
 97      1001  0001 ←— addition
               0110 ←— ajouter 6 pour corriger
      ----------------
      1001  0111      DCB de 97
        9     7
```

Dans ce cas-ci, l'addition des chiffres de poids le plus faible donne une somme de 17 = 10001. Il y a production d'un report sur le chiffre de rang immédiatement à gauche, report additionné avec les codes correspondant à 5 et à 3. Comme 17 > 9, on ajoute un facteur de correction de 6 à la somme des chiffres de poids faible. L'addition de cette correction ne donne pas lieu à un report; le report a déjà été produit au moment de la première addition.

Récapitulation de l'addition en DCB:

1. Addition binaire ordinaire des représentations DCB de tous les rangs.
2. Pour les rangs où la somme est égale ou inférieure à 9, aucune correction ne s'impose et la somme est une représentation DCB valide.
3. Quand la somme des deux chiffres est supérieure à 9, on ajoute une correction de 0110 pour obtenir la représentation DCB exacte. Il se produit toujours un report sur le chiffre de rang immédiatement à gauche, soit lors de l'addition initiale (première étape) ou de l'addition de la correction.

L'addition en DCB est de toute évidence plus compliquée que l'addition binaire courante. C'est également vrai pour les autres opérations arithmétiques en DCB. Effectuez par vous-même l'addition suivante: 275 + 641. Voici la réponse:

```
 275      0010 0111 0101 ←— DCB de 275
+641    + 0110 0100 0001 ←— DCB de 641
 916      1000 1011 0110 ←— addition
        +      0110      ←— ajouter 6 pour corriger le deuxième chiffre
          ---------------
          1001 0001 0110 ←— DCB de 916
```

QUESTIONS DE RÉVISION

1. Dites comment faire pour savoir s'il faut ajouter une correction à la réponse d'une addition en DCB.
2. Représentez 135_{10} et 265_{10} en DCB, puis additionnez-les. Vérifiez en reconvertissant en décimal.

6-8 ARITHMÉTIQUE HEXADÉCIMALE

Les nombres hexadécimaux sont utilisés en permanence dans la programmation en langage machine de même que pour désigner les adresses indiquant les places mémoires dans les

ordinateurs. Quand vous aurez à programmer en langage machine ou à manipuler des adresses de mémoire, il se peut que vous deviez additionner ou soustraire des nombres hexadécimaux.

Addition hexadécimale Ce type d'addition est effectué de la même façon que l'addition décimale, si on n'oublie pas que le chiffre le plus grand en hexadécimal est F plutôt que 9. La procédure proposée est celle-ci:

1. Additionner les deux chiffres hexadécimaux comme des chiffres décimaux, en remplaçant mentalement les lettres par leurs équivalents.
2. Si la somme est égale ou inférieure à 15, inscrire directement le chiffre hexadécimal.
3. Si la somme est égale ou supérieure à 16, soustraire 16 et reporter 1 sur le rang à gauche.

Voici des exemples:

EXEMPLE 6-6

Additionnez les nombres hexadécimaux 58 et 24.

Solution

$$\begin{array}{r} 58 \\ +\ 24 \\ \hline 7C \end{array}$$

L'addition des chiffres du rang de poids faible (8 et 4) donne 12, soit C en hexadécimal. Il n'y a pas de report sur le rang à gauche. L'addition de 5 et de 2 donne 7.

EXEMPLE 6-7

Additionnez les nombres hexadécimaux 58 et 4B.

Solution

$$\begin{array}{r} 58 \\ +\ 4B \\ \hline A3 \end{array}$$

Commençons par additionner 8 et B en remplaçant mentalement B par 11. Cela donne 19, soit une somme supérieure à 16; on soustrait donc 16 de 19, ce qui donne 3, que nous écrivons, et un report de 1 sur le rang immédiatement à gauche. Ce report est additionné à 5 et à 4 pour donner 10_{10}, soit le symbole A dans le système hexadécimal.

EXEMPLE 6-8

Additionnez 3AF et 23C.

Solution

$$\begin{array}{r} 3AF \\ +\ 23C \\ \hline 5EB \end{array}$$

La somme de F et C est $15 + 12 = 27_{10}$. Comme il s'agit d'une valeur supérieure à 16, soustrayons 16 de 27, ce qui donne 11_{10} soit l'équivalent de B dans le système hexadécimal et un report de 1 sur le second rang. L'addition de ce report avec A et 3 donne E, sans report sur le rang à gauche. Enfin, l'addition du rang de poids fort donne 5.

Soustraction hexadécimale Il faut se rappeler que les nombres hexadécimaux sont essentiellement une façon commode de représenter des nombres binaires. Par conséquent, on peut soustraire des nombres hexadécimaux en recourant à la même méthode que celle utilisée pour les nombres binaires. Le diminuteur hexadécimal est complémenté à 2 puis *additionné* au diminuande et tout report provenant de l'addition des chiffres de poids le plus fort est rejeté.

Comment trouvons-nous le complément à 2 d'un nombre hexadécimal? L'une des manières est de convertir le nombre en binaire, de prendre le complément à 2 de son équivalent binaire puis de reconvertir la réponse en hexadécimal. Voici un exemple de ceci.

73A	← nombre hexadécimal
0111 0011 1010	← conversion en binaire
1000 1100 0110	← complémenté à 2
8C6	← reconversion en hexadécimal

Il existe toutefois une marche à suivre beaucoup plus rapide. Soustraire *chaque* chiffre hexadécimal de F, puis ajouter 1. Faisons cela pour le même nombre hexadécimal que précédemment.

$$\begin{array}{rrr} F & F & F \\ -\ 7 & -\ 3 & -\ A \\ \hline 8 & C & 5 \\ & & +\ 1 \\ \hline 8 & C & 6 \end{array}$$

← soustraire chaque chiffre de F

← additionner 1

← l'équivalent hexadécimal du nombre complémenté à 2

Mettez en pratique l'une des deux marches à suivre précédentes pour trouver le complément à 2 du nombre hexadécimal E63. La réponse est 19D.

EXEMPLE 6-9

Soustrayez 3A5 de 592.

Solution D'abord trouvez le complément à 2 du diminuteur (3A5) au moyen de l'une des deux marches à suivre précédentes. La réponse est C5B. Ensuite additionnez ce chiffre au diminuande (592):

$$\begin{array}{r} 592 \\ +\ C5B \\ \hline 1ED \end{array}$$

↑ rejeté le report

Puisqu'on ne tient pas compte du report provenant de l'addition des chiffres du rang de poids le plus fort, la réponse est 1ED. Si on additionne 1ED à 3A5 on trouve 592, ce qui démontre bien que nous avons trouvé la bonne réponse.

QUESTIONS DE RÉVISION

1. Additionnez 67F + 2A4.
2. Soustrayez 67F − 2A4.

6-9 CIRCUITS ARITHMÉTIQUES

L'un des rôles essentiels des ordinateurs est d'effectuer des opérations arithmétiques. Dans un ordinateur, les opérations arithmétiques sont du ressort de l'unité arithmétique, où se trouvent des portes logiques et des bascules combinées de manière à additionner, soustraire, multiplier et diviser des nombres binaires. Ces circuits réalisent des opérations arithmétiques à des vitesses humainement impossibles à égaler. Habituellement une addition prend moins de 1 μs.

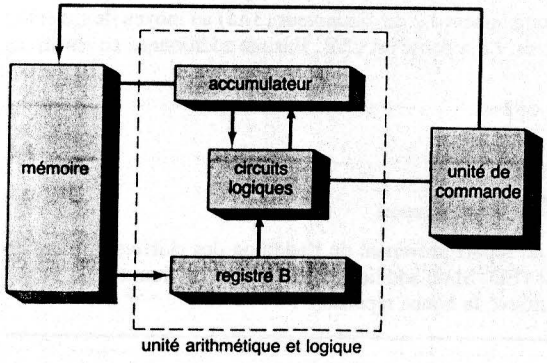
Nous allons maintenant étudier certains des circuits arithmétiques de base qui permettent d'effectuer les opérations arithmétiques examinées dans les sections précédentes. Dans quelques cas, nous allons voir tous les aspects de la conception réelle, et cela même si on trouve dans le commerce de tels circuits sous forme de circuits intégrés. Nous procédons ainsi en vue de vous donner une certaine pratique de l'utilisation des techniques apprises au chapitre 4.

Unité arithmétique et logique Toutes les opérations arithmétiques s'effectuent dans l'**unité arithmétique et logique** (UAL) de l'ordinateur. On peut voir à la figure 6-3 le schéma fonctionnel d'une unité arithmétique et logique type, illustrant ses principaux éléments. La principale raison d'être de l'unité UAL est d'accepter des données binaires stockées dans la mémoire et de les soumettre à des opérations arithmétiques conformément aux instructions reçues de l'unité de commande.

L'unité arithmétique et logique contient au moins deux registres de bascules: le **registre B** et le **registre accumulateur**. On y retrouve aussi des circuits logiques combinatoires qui opèrent sur les nombres binaires mémorisés dans le registre *B* et l'accumulateur pour donner un résultat arithmétique. Voici un exemple du déroulement des opérations:

1. L'unité de commande reçoit une instruction (de la mémoire) indiquant qu'un nombre mémorisé à un certain endroit de la mémoire (adresse) doit être additionné au nombre présentement contenu dans le registre accumulateur.

FIGURE 6-3 Éléments fonctionnels d'une unité arithmétique et logique.



2. Le nombre à additionner est transféré de la mémoire dans le registre *B*.
3. Le nombre se trouvant dans le registre *B* et le nombre du registre accumulateur sont additionnés par des circuits logiques (placés sous la régie de l'unité de commande). La somme résultante est mémorisée dans l'accumulateur.
4. Le nouveau nombre placé dans l'accumulateur peut rester là et être additionné à un autre nombre ou être transféré dans une place en mémoire, si le calcul arithmétique en question est terminé.

On voit d'après ce qui vient d'être dit pourquoi un des registres s'appelle accumulateur. La valeur contenue dans l'accumulateur grossit continuellement à mesure qu'on additionne à la somme qu'il contient d'autres nombres pris en mémoire. En fait, c'est dans l'accumulateur que sont versées les réponses des étapes intermédiaires d'un long calcul, de même que c'est dans lui qu'on place le résultat final du problème.

6-10 ADDITIONNEUR BINAIRE PARALLÈLE

Les ordinateurs ne peuvent additionner que deux nombres binaires à la fois, chacun de ces nombres binaires pouvant avoir plusieurs bits. La figure 6-4 illustre l'addition de deux nombres de 5 bits. Le *cumulande* est mémorisé dans l'accumulateur; donc les 5 bascules de l'accumulateur mémorisent les valeurs 10101. Le *cumulateur*, c'est-à-dire le nombre ajouté au cumulande, est mémorisé dans le registre B (dans ce cas-ci, 00111).

On commence l'addition en additionnant les bits de poids faible du cumulande et du cumulateur. Donc 1 + 1 = 10, ce qui signifie que la *somme* pour ce rang est 0 avec un *report* de 1.

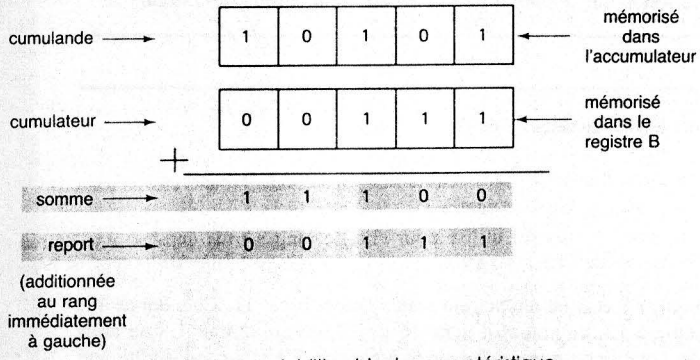


FIGURE 6-4 Addition binaire caractéristique.

Ce report est ajouté au bit du cumulande et du cumulateur du rang immédiatement à gauche. Ainsi l'addition au deuxième rang est 1 + 0 + 1 = 10, ce qui, à nouveau, produit une somme de 0 et un report de 1. Ce report est ajouté au bit du cumulateur et du cumulande du rang immédiatement à gauche et ainsi de suite pour les autres rangs.

À chaque étape de l'addition, nous additionnons 3 bits: le bit du cumulande, le bit du cumulateur et le bit de report provenant de l'addition des chiffres du rang précédent. Le résultat de l'addition de ces 3 bits est un nombre à 2 bits: le bit de *somme* et le bit de *report*, ce dernier devant être ajouté au rang immédiatement à gauche. Vous devez bien saisir que le même processus se répète pour tous les rangs du nombre. Donc, si on parvient à concevoir un circuit logique qui reproduit l'opération d'addition, il nous restera alors

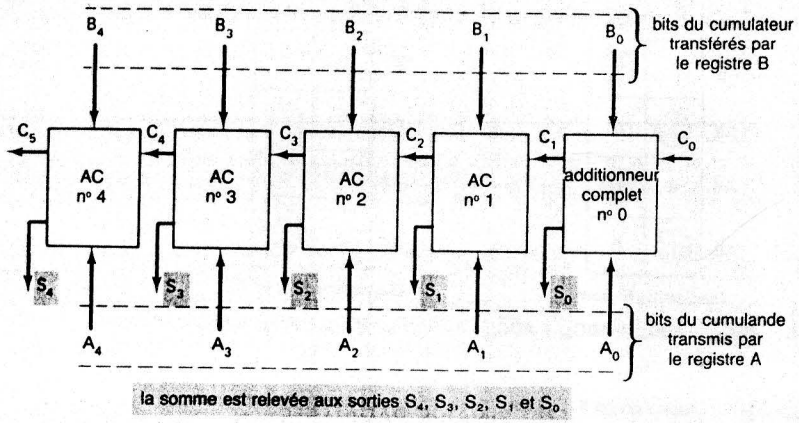


FIGURE 6-5 Schéma fonctionnel d'un circuit additionneur parallèle de 5 bits formé d'additionneurs complets.

simplement à accoler des circuits identiques, un pour chaque rang binaire. C'est ce que montre la figure 6-5.

Sur cette figure, les variables A_4, A_3, A_2, A_1 et A_0 représentent les bits du cumulande mémorisés dans l'accumulateur (appelé également registre A). Les variables B_4, B_3, B_2, B_1 et B_0 sont les bits du cumulateur mémorisés dans le registre B. Les variables C_4, C_3, C_2, C_1 et C_0 représentent les bits de report des rangs correspondants. Les variables S_4, S_3, S_2, S_1 et S_0 sont les bits de somme de chaque rang. Les bits correspondant du cumulateur et du cumulande sont appliqués à un circuit logique appelé **additionneur complet**, de même que le bit de report généré par l'addition des bits du rang précédent. Par exemple, les bits A_1 et B_1 sont transmis à l'additionneur complet 1 en même temps que C_1, qui est le bit de report produit par l'addition des bits A_0 et B_0. Les bits A_0 et B_0 sont fournis à l'additionneur complet 0 en même temps que C_0. Comme ces deux bits sont les bits de poids le plus faible du cumulateur et du cumulande, il est évident que C_0 doit toujours être nul, puisqu'il n'y a pas de rang précédent. Nous verrons toutefois qu'il y a des situations où C_0 peut être égal à 1.

L'additionneur complet utilisé pour chaque rang a trois entrées: une entrée A, une entrée B et une entrée C, et deux sorties: une sortie somme et une sortie report. Par exemple l'additionneur complet 0 a les entrées A_0, B_0 et C_0 et les sorties S_0 et C_1. L'additionneur complet 1 a les entrées A_1, B_1 et C_1 et les sorties S_1 et C_2; ainsi de suite pour les autres. Ce montage est répété pour le nombre de rangs qu'il y a dans le cumulande et le cumulateur. Sur cette figure, on additionne des nombres de 5 bits; dans les ordinateurs d'aujourd'hui les nombres s'échelonnent généralement de 8 à 64 bits.

Le montage de la figure 6-5 est appelé un **additionneur parallèle** parce que tous les bits du cumulateur et du cumulande sont appliqués et additionnés *simultanément*. Donc les additions des bits de chacun des rangs se font en même temps. Il s'agit là d'une façon qui diffère de la technique manuelle, dans laquelle on additionne chaque rang un à la fois en partant du bit de poids faible. De toute évidence l'addition parallèle est extrêmement rapide. Nous allons vous en dire plus dans les sections suivantes.

QUESTIONS DE RÉVISION

1. Dites combien un additionneur complet possède d'entrées. Dites combien il possède de sorties.

2. Supposez que les valeurs suivantes soient appliquées aux entrées de la figure 6-5: $A_4A_3A_2A_1A_0 = 01001$; $B_4B_3B_2B_1B_0 = 00111$; $C_0 = 0$.
   a) Dites quels sont les niveaux logiques des sorties de l'AC nº 2.
   b) Dites quel est le niveau logique de C_5.

6-11 CONCEPTION D'UN ADDITIONNEUR COMPLET

Maintenant que nous savons qu'elle est la fonction d'un additionneur complet, concevons un circuit logique qui effectue cette fonction. En premier lieu, nous devons construire une table de vérité énumérant toutes les combinaisons possibles pour les diverses valeurs d'entrée et de sortie. On peut voir cette table de vérité à la figure 6-6 dans laquelle se trouvent les trois entrées A, B et C_{en} et les deux sorties S et C_S. À trois entrées correspondent 8 combinaisons possibles; pour chaque combinaison, on donne dans la table les valeurs de sortie recherchées. Par exemple, dans le cas $A = 1$, $B = 0$ et $C_{en} = 1$, l'additionneur complet (abrégé en AC) doit additionner ces bits pour donner une somme (S) de 0 et un report (C_S) de 1. Le lecteur devrait vérifier les autres combinaisons afin de s'assurer qu'il comprend bien.

Comme il y a deux sorties, nous allons concevoir indépendamment les circuits de chacune des sorties, en commençant par celui de la sortie S. La table de vérité démontre qu'il y a 4 cas où S doit être 1. En recourant à la méthode des sommes de produits, nous pouvons écrire ainsi l'expression pour S:

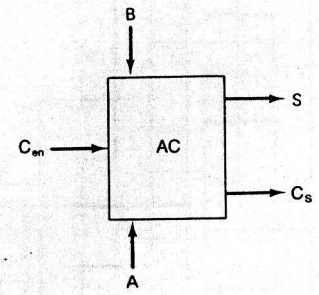
$$S = \overline{A}\overline{B}C_{en} + \overline{A}B\overline{C}_{en} + A\overline{B}\overline{C}_{en} + ABC_{en} \qquad (6\text{-}1)$$

Il est maintenant possible de simplifier cette expression en mettant en facteur. Malheureusement, il n'y a pas deux variables communes qui soient partagées par au moins deux autres termes de cette équation. Toutefois, il est possible de mettre en facteur $\overline{A}$ qui se retrouve dans les deux premiers termes et A qui se retrouve dans les deux derniers:

$$S = \overline{A}(\overline{B}C_{en} + B\overline{C}_{en}) + A(\overline{B}\overline{C}_{en} + BC_{en})$$

FIGURE 6-6 Table de vérité d'un additionneur complet.

Entrée du bit du cumulande	Entrée du bit du cumulateur	Entrée du bit de report	Sortie du bit de somme	Sortie du bit de report
A	B	C_{en}	S	C_S
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



On reconnaît que le premier terme entre parenthèses est l'équation du circuit OU exclusif pour B et C_{en}, que l'on écrit $B \oplus C_{en}$. Le second terme entre parenthèses est l'équation du circuit NI exclusif pour B et C_{en}, qui s'écrit $\overline{B \oplus C_{en}}$. Donc, l'expression pour S devient

$$S = \overline{A}(B \oplus C_{en}) + A\overline{(B \oplus C_{en})}$$

Soit $X = B \oplus C_{en}$, ce qui donne

$$S = \overline{A} \cdot X + A \cdot \overline{X} = A \oplus X$$

qui est simplement la mise en relation OU EXCLUSIF de A et X. En substituant à X sa valeur, nous obtenons

$$S = A \oplus [B \oplus C_{en}] \qquad (6\text{-}2)$$

Maintenant intéressons-nous à la sortie C_S de la table de vérité de la figure 6-6. L'expression de la somme de produits correspondant à C_S est:

$$C_S = \overline{A}BC_{en} + A\overline{B}C_{e-} + AB\overline{C}_{en} + ABC_{en}$$

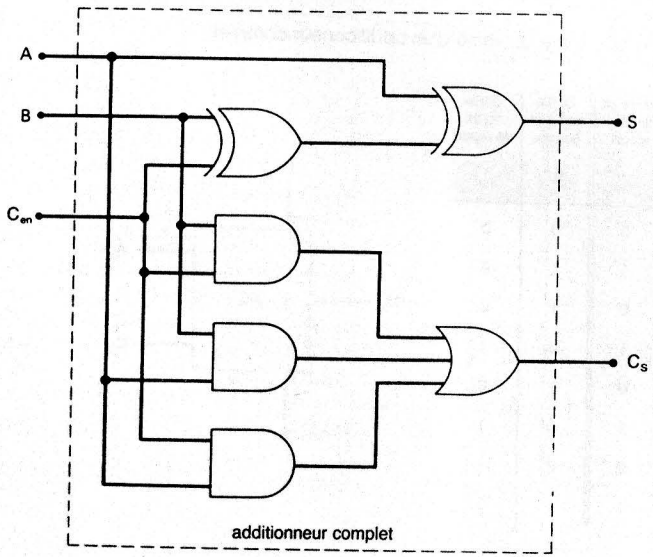
Cette expression peut être simplifiée en procédant à une mise en facteur. Nous allons appliquer l'artifice présenté au chapitre 3, c'est-à-dire additionner ABC_{en} *trois* fois puisqu'il a des facteurs communs avec chacun des autres termes. Nous avons donc,

$$\begin{aligned} C_S &= BC_{en}(\overline{A} + A) + AC_{en}(\overline{B} + B) + AB(\overline{C}_{en} + C_{en}) \qquad (6\text{-}3) \\ &= BC_{en} + AC_{en} + AB \end{aligned}$$

Cette expression ne peut être simplifiée plus avant.

Les expressions (6-2) et (6-3) sont matérialisées par le circuit de la figure 6-7. Il existe plusieurs autres circuits qui donnent les mêmes expressions pour S et C_S, quoi-

FIGURE 6.7 Ensemble des circuits d'un additionneur complet.



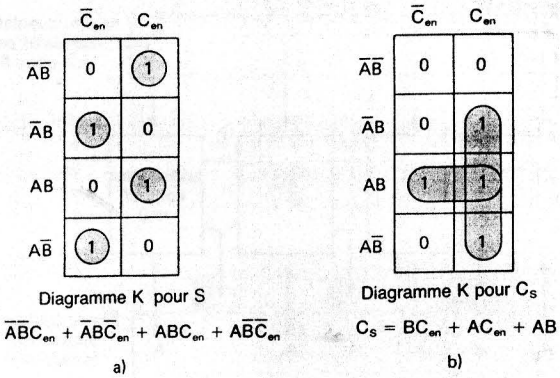


FIGURE 6-8 Diagrammes K pour les sorties d'un additionneur complet.

qu'aucun ne soit vraiment plus avantageux que celui illustré. Le circuit intégral avec ses entrées A, B et C_{en} et ses sorties S et C_S représente un additionneur complet. Chacun des AC de la figure 6-5 est identique à ce circuit.

Simplification avec les diagrammes K Nous avons simplifié les expressions pour S et C_S à l'aide des méthodes algébriques. On peut également recourir à la méthode des diagrammes de Karnaugh. La figure 6-8 a) nous fait voir le diagramme de Karnaugh pour la sortie S. On ne voit aucun 1 adjacent dans ce diagramme, il n'y a donc pas de doublets ou de quartets à réunir. Donc l'expression pour S ne peut être simplifiée au moyen d'un diagramme K. On a donc ici un exemple où la méthode du diagramme K ne donne aucun résultat contrairement à la méthode algébrique. Nous avons été en mesure de simplifier l'expression pour S en effectuant des mises en facteur et en recourant aux opérations OU EXCLUSIF et NI EXCLUSIF.

Le diagramme K de la sortie C_S est montré à la figure 6-8 b). Les trois doublets réunis permettent d'obtenir la même expression que celle trouvée au moyen de la méthode algébrique.

Le demi-additionneur L'additionneur complet transforme trois entrées pour donner une sortie somme et une sortie report. Dans certains cas, on a besoin d'un circuit qui additionne seulement deux bits d'entrée pour donner un bit somme et un bit de report. Un exemple de ceci est l'addition des deux bits de poids faible quand il n'y a pas de report à considérer. Il est possible de concevoir un circuit logique qui opère sur *deux* bits d'entrée, A et B, pour produire une sortie somme (S) et une sortie report (C_S). Ce circuit s'appelle un **demi-additionneur** (DA). Son fonctionnement est analogue à celui de l'additionneur complet, sauf que son action ne porte que sur deux bits. Nous laissons à l'étudiant le soin de concevoir le DA dans un des problèmes de la fin du chapitre (problème 6-18). Toutefois, nous n'hésiterons pas à utiliser le symbole du DA lorsque ce sera nécessaire.

6-12 ADDITIONNEUR PARALLÈLE COMPLET AVEC SES REGISTRES

Dans un ordinateur, les nombres à additionner sont mémorisés dans des registres à bascules. On a reproduit à la figure 6-9 le schéma complet d'un additionneur parallèle de 4 bits avec ses registres de stockage. Les bits du cumulande A_3–A_0 sont mémorisés dans l'accumulateur

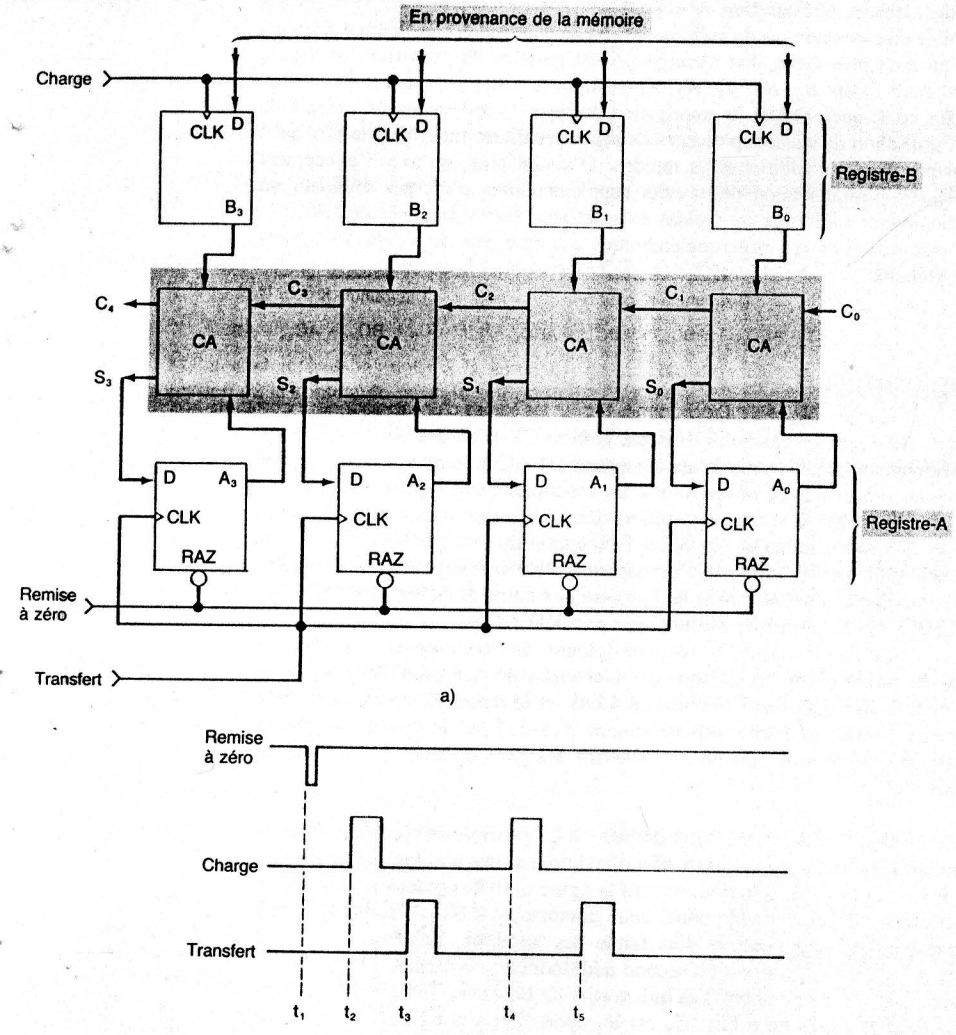


FIGURE 6-9 a) Un additionneur 4 bits parallèle complet; b) les signaux qui commandent l'addition de nombres binaires provenant de la mémoire dont la somme est placée dans la mémoire.

(registre A); les bits du cumulateur B_3–B_0 sont conservés dans le registre B. Chacun de ces registres est formé de bascules D afin de faciliter le transfert des données.

Le contenu du registre A (c'est-à-dire le nombre binaire mémorisé dans A_3–A_0) est ajouté au contenu du registre B par les 4 additionneurs complets, et la somme apparaît sur les sorties S_3–S_0. C_4 est le report du quatrième additionneur complet, et on peut l'appliquer à l'entrée report d'un cinquième additionneur complet ou le considérer comme un bit *de dépassement* afin d'indiquer que la somme excède 1111.

Notez que les sorties de la somme sont raccordées aux entrées D du registre A. De cette manière la somme est transférée en parallèle dans le registre A au moment du FM de l'impulsion transfert. Au total, la somme se retrouve mémorisée dans le registre A.

En outre, remarquez que les entrées D du registre B sont alimentées par des bits provenant de la mémoire de l'ordinateur; les nombres provenant de la mémoire sont transférés en parallèle dans le registre B au moment du FM de l'impulsion CHARGE. Pour l'instant, gardons sous silence la façon selon laquelle les nombres arrivent de la mémoire.

Convention d'écriture des registres Avant de voir en détails comment ce circuit additionne des nombres binaires, il est bon de vous indiquer une certaine convention d'écriture qu'on utilise pour décrire facilement le contenu d'un registre et les opérations de transfert de données.

Toutes les fois que nous voulons indiquer les niveaux qui se trouvent dans chaque bascule d'un registre ou sur chaque sortie d'un groupe, nous utiliserons les crochets; voici un exemple:

$$[A] = 1011$$

Cette façon d'écrire est équivalente à dire $A_3 = 1, A_2 = 0, A_1 = 1, A_0 = 1$. Autrement dit, chaque fois que vous voyez $[A]$, vous devez traduire par « le contenu du registre A ».

Chaque fois que nous voulons signaler un transfert de données entre deux registres, nous utilisons une flèche comme ci-dessous:

$$[B] \longrightarrow [A]$$

Cette convention d'écriture signifie que le contenu du registre B a été transféré dans le registre A. À la suite de cette opération, ce qui se trouvait dans le registre A est changé et ce qui se trouvait dans le registre B s'y trouve encore.

Séquence des opérations Décrivons maintenant l'addition des deux nombres binaires 1001 et 0101 par le circuit de la figure 6-9. Supposons $C_0 = 0$; c'est-à-dire qu'il n'y a pas de report sur le rang de poids le plus faible.

1. **$[A] = 0000$.** Une impulsion REMISE A ZÉRO est appliquée aux entrées asynchrones (*RAZ*) de chaque bascule du registre A.
2. **$[M] \rightarrow [B]$.** Le premier nombre binaire est transféré de la mémoire (M) dans le registre B. Dans ce cas, le nombre binaire 1001 est placé dans le registre B à l'arrivée du FM de l'impulsion CHARGE en t_2.
3. **$[S]^* \rightarrow [A]$.** Quand $[B] = 1001$ et $[A] = 0000$, les additionneurs complets produisent une somme de 1001, c'est-à-dire $[S] = 1001$. Les sorties de la somme sont transférées dans le registre A à l'arrivée du FM de l'impulsion TRANSFERT en t_3. On se retrouve donc avec $[A] = 1001$.
4. **$[M] \rightarrow [B]$.** Le second nombre binaire, 0101, est transféré de la mémoire dans le registre B à l'arrivée du FM de la deuxième impulsion CHARGE en t_4. Ceci donne $[B] = 0101$.
5. **$[S] \rightarrow [A]$.** Quand $[B] = 0101$ et $[A] = 1001$, les additionneurs complets donnent $[S] = 1110$. Les sorties de la somme sont transférées dans le registre A à l'arrivée du FM de la deuxième impulsion TRANSFERT en t_5. Donc $[A] = 1110$.

* Même si S n'est pas un registre, nous utilisons la notation $[S]$ pour représenter le groupe des sorties S.

6. À cet instant, la somme des deux nombres binaires se trouve mémorisée dans l'accumulateur. Dans la plupart des ordinateurs, le contenu de l'accumulateur, soit [*A*], est le plus souvent transféré dans une place mémoire de manière à libérer l'additionneur pour qu'il serve à d'autres ensembles de nombres. Les circuits qui assurent le transfert [*A*] → [*M*] ne sont pas illustrés sur ce schéma.

QUESTIONS DE RÉVISION

1. Supposez que vous retirez de la mémoire 4 nombres différents de 4 bits et que vous les additionnez au moyen du circuit de la figure 6-9. Dites combien il faudra d'impulsions REMISE À ZÉRO. Dites combien il faudra d'impulsions CHARGE. Dites combien il faudra d'impulsions TRANSFERT.
2. Indiquez le contenu du registre *A* après la séquence des opérations suivante: [*A*] = 0000, [0110] → [*B*], [*S*] → [*A*], [1110] → [*B*], [*S*] → [*A*].

6-13 PROPAGATION DU REPORT

L'additionneur parallèle de la figure 6-9 additionne très rapidement, puisqu'il traite les bits de tous les rangs simultanément. Toutefois, sa vitesse est limitée par un phénomène appelé **propagation du report** ou **répercussion du report**, dont vous comprendrez mieux le mode d'action en étudiant l'addition suivante :

$$\begin{array}{r} 0111 \\ +\ 0001 \\ \hline 1000 \end{array}$$

L'addition des chiffres du rang de poids faible donne un report ramené sur le deuxième rang. Ce report, lorsqu'il est additionné aux bits du second rang, donne lieu à un report ramené sur le troisième rang. Ce dernier report, quand il est additionné aux bits du troisième rang, donne lieu à un report ramené sur le dernier rang. Ce qu'il faut avant tout remarquer dans cet exemple c'est que le bit de somme généré par l'addition des bits du *dernier* rang (poids fort) est conditionné par le report généré lors de l'addition des bits du *premier* rang (poids faible).

Vu de cet angle, le circuit de la figure 6-9 est un montage dont la sortie S_3 du dernier additionneur complet est fonction de la sortie C_1 du premier additionneur. Il faut que la valeur sur C_1 se propage à travers les trois autres AC avant de fixer la valeur S_3. Cela veut donc dire que la sortie S_3 ne correspond pas à la valeur exacte de l'addition tant que la valeur de C_1 n'a pas été traitée par les AC intermédiaires. Il y a donc là un retard qui dépend de la vitesse de propagation dans les AC. Par exemple, si chaque AC présente un retard de propagation de 40 ns, S_3 n'affichera le niveau exact que 120 ns après la production de C_1. L'impulsion de commande TRANSFERT ne peut donc être appliquée que 160 ns après le transfert dans les registres à bascules du cumulande et du cumulateur (les 40 ns supplémentaires s'expliquent par le retard du AC de poids faible qui produit C_1).

De toute évidence, cette situation empire si on ajoute d'autres additionneurs afin de trouver la somme de nombres ayant plus de rangs. Si notre additionneur pouvait opérer sur des nombres de 32 bits, le retard de propagation du report sera de 1280 ns = 1,28 μs. L'impulsion de transfert ne pourra être appliquée, au plus tôt, que 1,28 μs après le transfert des nombres dans les registres.

Un retard de cet ordre est inacceptable pour les ordinateurs rapides. Heureusement, l'ingéniosité des concepteurs a permis de trouver des solutions qui réduisent ce retard.

Dans l'une d'elles, appelée **anticipation du report**, des portes logiques examinent les bits de poids plus faible du cumulateur et du cumulande afin de déterminer s'il y aura génération de reports sur un rang plus élevé. Par exemple, il est possible de construire un circuit logique dont les entrées sont B_2, B_1, B_0, A_2, A_1 et A_0 et la sortie est C_3. Le retard d'un tel circuit est plus court que celui de la propagation du report à travers les AC. Une telle solution exige l'utilisation de beaucoup de circuits supplémentaires mais c'est le prix qu'il faut payer pour produire des additionneurs rapides. D'autant plus, qu'avec l'avènement des circuits intégrés, l'implantation de circuits supplémentaires n'est pas vraiment un problème. De nombreux additionneurs rapides existant sous forme intégrée exploitent la technique de l'anticipation du report ou une technique analogue afin de réduire les retards de propagation globaux.

6-14 ADDITIONNEUR PARALLÈLE INTÉGRÉ

Il existe plusieurs additionneurs parallèles dans des boîtiers CI. L'un des plus courants est un boîtier d'additionneurs parallèles de 4 bits comprenant 4 AC et leurs connexions et les circuits de l'anticipation du report nécessaires à un fonctionnement rapide. Les 7483A, 74LS83A, 74283 et 74LS283 sont tous des boîtiers d'additionneurs parallèles 4 bits de la famille TTL. Les 283 sont identiques aux 83, à l'exception du fait que V_{CC} et la masse sont respectivement sur les broches 16 et 8; c'est désormais la norme sur toutes les nouvelles puces d'avoir les broches d'alimentation et de la masse aux coins du boîtier. Le 74HC283 est la version CMOS rapide du même additionneur parallèle 4 bits.

On peut voir à la figure 6-10 a) le symbole logique de l'additionneur parallèle de 4 bits 74LS83. Les entrées de ce CI sont deux groupes de 4 bornes $A_3A_2A_1A_0$ et $B_3B_2B_1B_0$, servant à appliquer deux nombres de 4 bits, et le report C_0 appliqué au rang de poids faible. Les sorties sont les 4 bits de somme $S_3S_2S_1S_0$ et le report C_4 quittant l'étage de poids fort. Les bits de somme sont souvent désignés $\Sigma_3\Sigma_2\Sigma_1\Sigma_0$ où Σ est la lettre grecque *sigma*.

Montage en cascade d'additionneurs parallèles Il est possible de raccorder deux ou plusieurs additionneurs parallèles en cascade afin d'additionner des nombres ayant un plus grand nombre de bits. À titre d'illustration, voyons la figure 6-10 b) où deux additionneurs 74LS83 sont reliés afin de pouvoir additionner deux nombres de 8 bits. L'additionneur de droite additionne les 4 bits de poids le plus faible des nombres. La sortie C_4 de cet additionneur est reliée à l'entrée report du second additionneur; ce dernier additionne les 4 bits de poids le plus fort des nombres. Les huit sorties de la somme indiquent le résultat de l'addition de deux nombres de 8 bits. C_8 est le report fourni par l'étage du rang de poids le plus fort du second additionneur. Cette sortie peut devenir un bit de dépassement ou servir de report pour un autre étage d'additionneur, si on traite des nombres binaires encore plus longs.

EXEMPLE 6-10

Déterminez les niveaux logiques aux entrées et aux sorties de l'additionneur 8 bits de la figure 6-10 b) quand 137_{10} est additionné à 72_{10}.

Solution Première étape, convertissons chaque nombre en son équivalent binaire de 8 bits:

$$\begin{aligned} 137 &= 10001001 \\ 72 &= 01001000 \end{aligned}$$

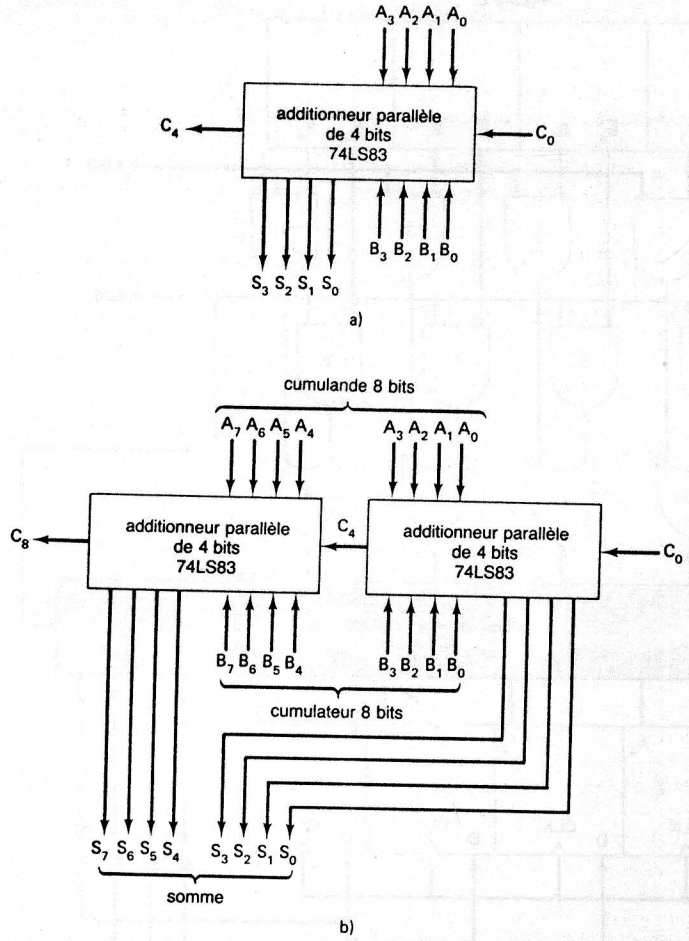


FIGURE 6-10 a) Symbole rectangulaire de l'additionneur parallèle de 4 bits 74LS83; b) deux 74LS83 montés en cascade.

Ces deux expressions binaires sont appliquées aux entrées A et B; c'est-à-dire on retrouve sur les entrées A de gauche à droite 10001001, et sur les entrées B, aussi de gauche à droite, 01001000. L'additionneur effectue l'addition binaire de ces deux nombres:

$$\begin{aligned}[A] &= 10001001\\ [B] &= \underline{01001000}\\ [S] &= 11010001\end{aligned}$$

Les sorties de la somme portent de gauche à droite le chiffre 11010001. Il n'y a pas de dépassement sur le bit C_8, qui est donc un 0.

6-15 NOTATION COMPLÉMENT À 2

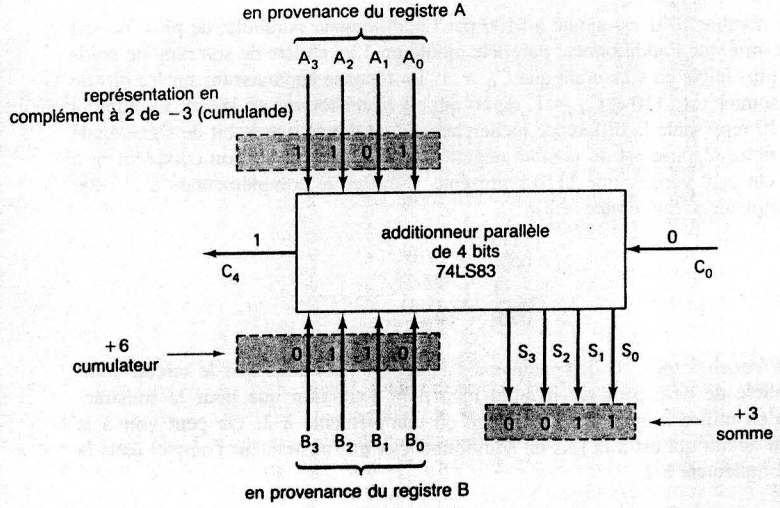
Aujourd'hui, la majorité des ordinateurs expriment leurs nombres négatifs selon la notation en complément à 2 et utilisent cette notation quand ils effectuent des soustractions. L'addition et la soustraction des nombres signés se résument à une simple addition si on exprime les nombres négatifs selon la notation en complément à 2.

Addition L'addition de nombres positifs et négatifs, y compris leur bit de signe, est possible dans l'additionneur parallèle de base si les nombres négatifs sont sous leur version complément à 2. On peut voir une illustration de ceci à la figure 6-11 dans le cas de l'addition de -3 à $+6$. Le nombre -3 est sous sa forme complément à 2 soit 1101, où le premier 1 est le bit de signe; le nombre $+6$ est sous sa forme normale, soit 0110, le premier 0 étant le bit de signe. Ces nombres sont mémorisés dans les registres correspondants. L'additionneur parallèle de 4 bits produit alors la somme 0011 sur les bornes de sortie, ce qui correspond à $+3$. La sortie C_4 est 1, mais ce report est rejeté dans la méthode complément à 2.

Soustraction Quand on soustrait en utilisant la notation complément à 2, le diminuteur est complémenté à 2 puis *additionné* au diminuande (le nombre dont on soustrait le diminuteur). Supposons que le diminuande est déjà mémorisé dans l'accumulateur (registre A). Le diminuteur se voit alors transféré dans le registre B (dans un ordinateur ce nombre provient de la mémoire) puis est complémenté à 2 avant d'être additionné au nombre contenu dans le registre A. Les sorties somme de l'additionneur représentent la *différence* entre le diminuande et le diminuteur.

Le circuit additionneur parallèle déjà examiné peut être modifié pour soustraire si on prévoit une façon de prendre le complément à 2 du nombre contenu dans le registre B. On sait que le complément à 2 d'un nombre binaire s'obtient en complémentant (inversant)

FIGURE 6-11 Utilisation d'un additionneur parallèle pour additionner des nombres positifs et négatifs dans la notation en complément à 2.



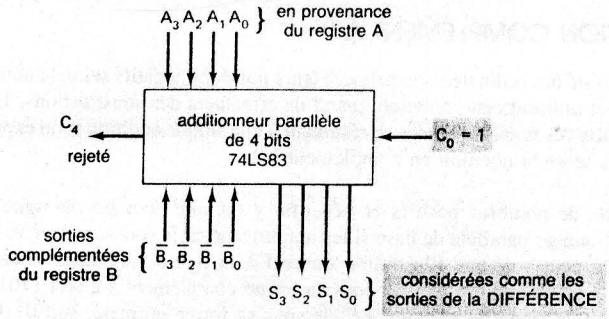


FIGURE 6-12 Additionneur parallèle servant à soustraire

chaque bit et en ajoutant 1 à son bit de poids le plus faible. La figure 6-12 montre comment un additionneur peut servir pour la soustraction. On utilise les sorties *complémentées* du registre B plutôt que les sorties normales; c'est-à-dire qu'on applique aux entrées de l'additionneur $\overline{B}_0$, $\overline{B}_1$, $\overline{B}_2$ et $\overline{B}_3$ (se rappeler que B_3 est le bit de signe). De cette manière, on complémente chaque bit du nombre B. En outre, on transforme C_0 en un 1 logique, de sorte qu'on additionne ainsi un 1 au bit de poids le plus faible de l'additionneur; cette façon de procéder a le même effet que d'additionner 1 au bit de poids le plus faible du registre B afin d'obtenir son complément à 2.

Les sorties S_3–S_0 présentent un nombre correspondant au résultat de l'opération de soustraction. Évidemment S_3 est le bit de signe du résultat et indique qu'il est soit positif, soit négatif. Le report C_4 est à nouveau rejeté.

Pour bien fixer les idées, étudions les diverses étapes de la soustraction suivante: $+4 - (+6)$:

1. $+4$ est mémorisé dans le registre A sous la forme 0100.
2. $+6$ est mémorisé dans le registre B sous la forme 0110.
3. Les sorties complémentées des bascules du registre B sont transmises à l'additionneur, ce qui arrive est donc 1001.
4. Le nombre 1001 est ajouté à 0100 par l'additionneur parallèle; de plus, on fait en sorte que l'additionneur parallèle ajoute un 1 au chiffre de son rang de poids le plus faible en s'assurant que $C_0 = 1$. La réponse apparaissant sur les bits de la somme est 1110 et $C_4 = 1$, report qui est rejeté comme on le sait. Ce nombre 1110 représente la différence recherchée. Etant donné que le bit de signe égale 1, notre réponse est un nombre négatif exprimé dans sa version complément à 2. On peut vérifier que 1110 représente -2_{10} en le complémentant à 2; cette complémentation donne $+2_{10}$:

$$\begin{array}{r} 1110 \\ 0001 \\ +\quad 1 \\ \hline 0010 = +2_{10} \end{array}$$

Additionneur/soustracteur Il doit maintenant être clair pour tous que le circuit additionneur parallèle de base peut servir aussi bien pour l'addition que pour la soustraction selon qu'on utilise le nombre B inchangé ou complémenté à 2. On peut voir à la figure 6-13 un circuit qui est à la fois un additionneur et un soustracteur complet dans la notation en complément à 2.

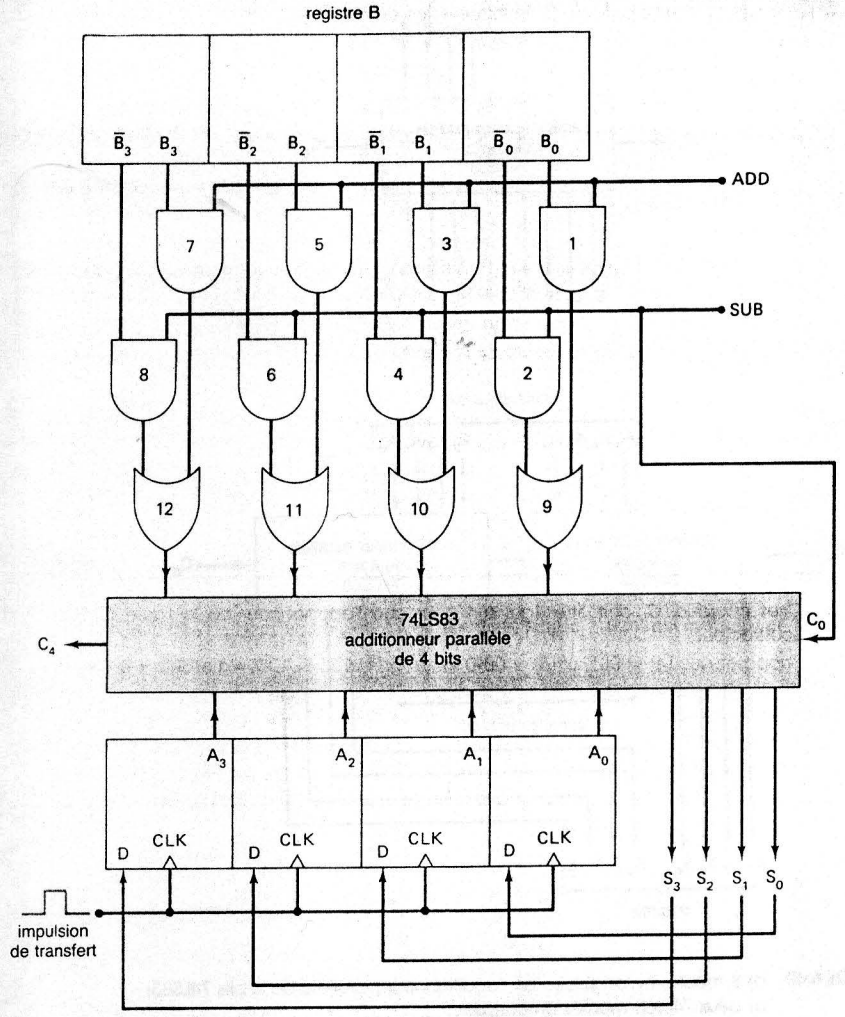


FIGURE 6-13 Additionneur/soustracteur parallèle dans la notation en complément à 2.

Cet additionneur/soustracteur est commandé par deux signaux de commande ADD et SUB. Quand le niveau ADD est HAUT, le circuit fonctionne comme un additionneur des nombres mémorisés dans les registres A et B. Quand le niveau SUB est HAUT, le circuit fonctionne comme un soustracteur qui retranche du nombre se trouvant dans A le nombre se trouvant dans B. Voici une description de son fonctionnement:

1. Supposons que ADD = 1 et SUB = 0. La commande SUB = 0 *invalide* (bloque) les portes ET 2, 4, 6 et 8, gardant ainsi leurs sorties à 0. La commande ADD = 1 *valide* les portes ET 1, 3, 5 et 7, ce qui autorise le transfert à travers ces portes des niveaux sur les bornes B_0, B_1, B_2 et B_3 respectivement.

2. Les niveaux B_0–B_3 franchissent les portes OU et arrivent dans l'additionneur parallèle de 4 bits en vue d'être additionnés aux bits A_0–A_3. La *somme* apparaît sur les sorties S_0–S_3.
3. Notez que SUB = 0 fait en sorte que C_0 de l'additionneur = 0.
4. Supposons maintenant que ADD = 0 et SUB = 1. La commande ADD = 0 bloque les portes ET 1, 3, 5 et 7. La commande SUB = 1 valide les portes ET, 2, 4, 6 et 8 de sorte que les niveaux sur $\overline{B}_0$, $\overline{B}_1$, $\overline{B}_2$ et $\overline{B}_3$ traversent respectivement ces quatre portes.
5. Les niveaux $\overline{B}_0$–$\overline{B}_3$ sont communiqués à travers les portes OU jusqu'à l'additionneur en vue d'être additionnés aux bits A_0–A_3. Remarquez qu'à présent C_0 vaut 1. Ainsi, le nombre contenu dans le registre B a bel et bien été complémenté à 2.
6. La *différence* se retrouve sur les sorties S_0–S_3.

Le montage de la figure 6-13 est l'un des plus répandus dans les ordinateurs, parce que c'est une solution relativement simple pour l'addition et la soustraction de nombres négatifs et positifs. Dans la plupart des ordinateurs, les niveaux présents sur les lignes de sorties S sont retournés au registre A (accumulateur); donc on retrouve toujours dans ce registre le résultat de l'addition ou de la soustraction. Ce transfert est commandé par une impulsion sur les entrées *CLK* du registre A.

QUESTIONS DE RÉVISION

1. Dites pourquoi C_0 doit être égal à 1 pour que l'additionneur de la figure 6.12 soit un soustracteur.
2. Supposez que [A] = 0011 et [B] = 0010 sur la figure 6-13. Si ADD = 1 et SUB = 0, déterminez les niveaux logiques des sorties des portes OU.
3. Même question qu'en 2 mais cette fois-ci pour ADD = 0 et SUB = 1.
4. *Vrai ou faux:* Quand le circuit additionneur/soustracteur sert pour la soustraction, c'est le complément à 2 du diminuande qui est aux bornes d'entrée de l'additionneur.

6-16 ADDITIONNEUR DCB

La marche à suivre pour additionner en DCB a été analysée à la section 6-7; revoyons-la succinctement:

1. Additionnez les représentations codées DCB pour chaque rang du chiffre décimal; utilisez les règles de l'addition binaire ordinaire.
2. Pour les rangs du nombre dont la somme est inférieure à 9, la réponse est déjà une représentation DCB valide et aucune correction n'est nécessaire.
3. Quand la somme de deux chiffres dépasse 9, il faut additionner une correction de 0110 à la somme pour obtenir la représentation codée DCB valide. Ceci a pour effet de produire un report qui doit être additionné au rang immédiatement à gauche du nombre décimal.

Un additionneur DCB doit pouvoir matérialiser les étapes précédentes. Autrement dit, ce circuit doit être en mesure de:

1. Additionner deux représentations codées DCB de 4 bits selon les règles ordinaires de l'addition binaire.

2. Déterminer si la somme est supérieure à 1001 (chiffre décimal 9); dans l'affirmative, ajouter 0110 (6) à cette somme et générer un report ramené sur le rang décimal immédiatement à gauche.

L'exigence 1. ne pose pas vraiment de problème puisqu'on la réalise au moyen d'un additionneur parallèle binaire 4 bits comme le CI 74LS83. Par exemple, si deux représentations codées DCB se trouvant respectivement sur $A_3A_2A_1A_0$ et $B_3B_2B_1B_0$ sont appliquées à un additionneur parallèle de 4 bits, ce dernier réalise l'opération suivante:

$$\begin{array}{rl} A_3A_2A_1A_0 & \longleftarrow \text{ représentation codée DCB} \\ +\ B_3B_2B_1B_0 & \longleftarrow \text{ représentation codée DCB} \\ \hline S_4S_3S_2S_1S_0 & \longleftarrow \text{ somme binaire courante} \end{array}$$

S_4 est en réalité C_4, le report produit par le rang du bit de poids le plus fort.

Les sorties de la somme $S_4S_3S_2S_1S_0$ peuvent afficher toutes les valeurs de l'intervalle 00000 à 10010 (quand les deux représentations sont égales à 1001 = 9). On doit retrouver dans l'additionneur DCB des portes logiques qui décident quand $S_4S_3S_2S_1S_0$ est supérieur à 01001 et commandent l'addition de la correction. La somme dépasse 01001 dans les cas énumérés ci-après:

Soit X une sortie logique qui occupera le niveau HAUT seulement quand la somme est supérieure à 01001 (c'est-à-dire dans les cas énumérés ci-après). L'étude de ces différents cas nous permet de découvrir que X est au niveau HAUT quand l'une ou l'autre des conditions suivantes est réalisée:

1. Quand S_4 = 1 (somme supérieure à 15).
2. Quand S_3 = 1 et que S_2 ou S_3 ou les deux sont à 1 (somme entre 10 et 15).

S_4	S_3	S_2	S_1	S_0	
0	1	0	1	0	(10)
0	1	0	1	1	(11)
0	1	1	0	0	(12)
0	1	1	0	1	(13)
0	1	1	1	0	(14)
0	1	1	1	1	(15)
1	0	0	0	0	(16)
1	0	0	0	1	(17)
1	0	0	1	0	(18)

Formulée sous la forme d'une équation booléenne, cette constatation est:

$$X = S_4 + S_3(S_2 + S_1)$$

Quand X = 1, il faut additionner la correction 0110 aux bits de la somme et produire un report. La figure 6-14 nous montre les circuits complets d'un additionneur DCB, y compris les portes de décision logiques qui produisent X.

Ce circuit est constitué de trois parties fondamentales. Les deux représentations codées DCB A_3–A_0 et B_3–B_0 sont additionnées dans l'additionneur 4 bits du haut pour donner la somme S_4 S_3 S_2 S_1 S_0; les portes logiques matérialisent la condition X. L'additionneur 4 bits du bas additionne la correction 0110 aux bits de somme seulement quand X = 1 pour donner la somme DCB représentée par $\Sigma_3\Sigma_2\Sigma_1\Sigma_0$. X est également utilisé comme report; celui qu'il faut produire quand la somme dépasse 01001. Évidemment quand X = 0, il n'y a pas de report et pas d'addition de 0110. Dans de tels cas, $\Sigma_3\Sigma_2\Sigma_1\Sigma_0 = S_3S_2S_1S_0$.

Nous recommandons au lecteur qui veut approfondir sa connaissance de l'additionneur DCB de faire plusieurs additions en suivant le cheminement des bits dans le circuit. À cet égard, les exemples suivants peuvent être particulièrement instructifs:

entrées

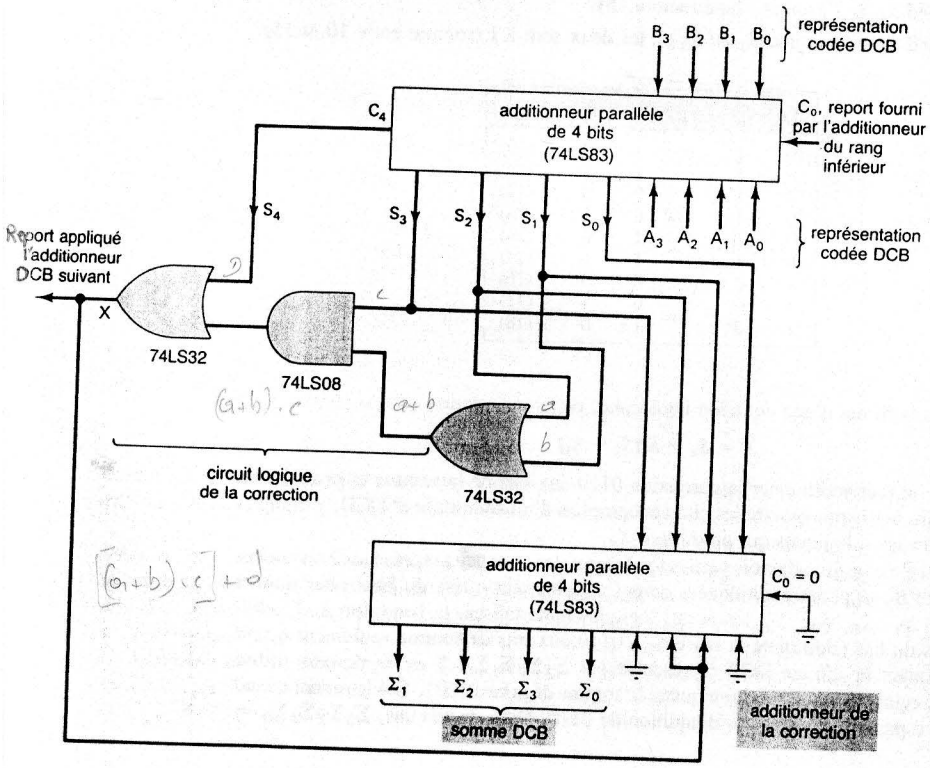
a) $[A] = 0101, [B] = 0011, C_0 = 0$
b) $[A] = 0111, [B] = 0110, C_0 = 0$

sorties

a) $[S] = 01000, X = 0, [\Sigma] = 1000$, REPORT = 0
b) $[S] = 01101, X = 1, [\Sigma] = 0011$, REPORT = 1

Additionneurs DCB montés en cascade Le circuit de la figure 6-14 sert seulement à additionner deux chiffres décimaux codés en DCB. Quand les nombres décimaux possèdent plusieurs chiffres, il faut pour leur addition utiliser un additionneur DCB distinct pour chaque rang de chiffres dans le nombre. La figure 6-15 illustre l'addition de deux nombres

FIGURE 6-14 Additionneur DCB contenant deux additionneurs de 4 bits et un circuit de détection de la correction.



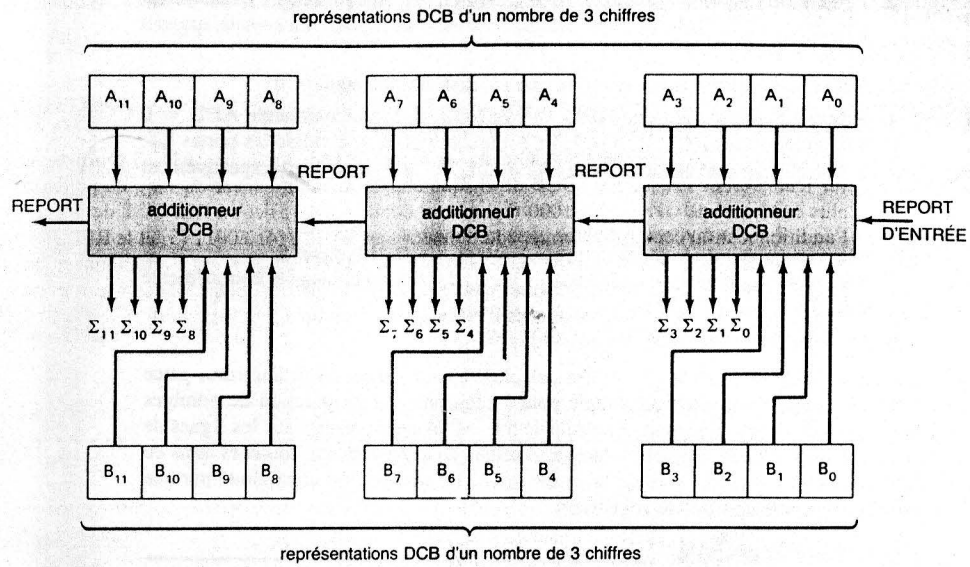


FIGURE 6-15 Additionneurs DCB montés en cascade pour additionner deux nombres décimaux de trois chiffres.

décimaux de trois chiffres. Le registre A contient 12 bits, ce qui est suffisant pour 3 représentations codées DCB d'un nombre décimal de 3 chiffres; de même le registre B contient la représentation DCB de l'autre nombre décimal de 3 chiffres. Les représentaitons codées A_3–A_0 et B_3–B_0 correspondent aux chiffres de poids faible qui sont appliqués au premier additionneur DCB. Chaque étage d'additionneurs DCB contient tous les circuits illustrés à la figure 6-14. Le premier additionneur DCB produit le groupe des sorties $\Sigma_3\Sigma_2\Sigma_1\Sigma_0$, la représentation DCB du chiffre de poids faible de la somme. Le report de ce premier additionneur est envoyé au second additionneur DCB qui additionne A_7–A_4 à B_7–B_4, les représentations codées DCB des chiffres décimaux du second rang du nombre. Ce deuxième additionneur produit $\Sigma_7\Sigma_6\Sigma_5\Sigma_4$, la représentation DCB du second chiffre de la somme, et de même pour le troisième chiffre. Évidemment il est possible d'étendre ce montage pour additionner des nombres décimaux ayant une longueur quelconque simplement en ajoutant des bascules aux registres et en affectant un additionneur DCB à chaque rang supplémentaire.

EXEMPLE 6-11

Déterminez les entrées et les sorties du circuit de la figure 6-15 si on l'utilise pour additionner 247_{10} et 538_{10}.

Solution Tout d'abord, exprimons les nombres décimaux en code DCB.

$$247 = 0010\ 0100\ 0111 \text{ (DCB)}$$
$$538 = 0101\ 0011\ 1000 \text{ (DCB)}$$

Ces représentations DCB sont placées dans les registres A et B, respectivement, d'où:

$$[A] = 0010\ 0100\ 0111$$

$$[B] = 0101\ 0011\ 1000$$

Le REPORT D'ENTRÉE qui arrive de l'addition du rang de poids le plus faible est 0.

Une fois que les données sont dans les registres, les additionneurs DCB fournissent sur leurs sorties les sommes DCB de chaque chiffre. L'additionneur du rang de poids le plus faible ajoute 0111 (7) à 1000 (8), ce qui donne 0101 (5) et un REPORT de 1 pour l'additionneur du centre. Ce dernier additionneur ajoute 0100 (4) à 0011 (3) et le REPORT de 1, ce qui produit une somme de 1000 (8) et un REPORT de 0 sur l'additionneur du rang de poids le plus fort. Ce dernier ajoute 0010 (2) à 0101 (5) pour donner une somme de (7) et aucune REPORT en sortie. Les sorties de somme s'affichent alors:

$$[\Sigma] = 0111\ 1000\ 0101$$

et le REPORT de l'étage du rang de poids le plus fort est 0.

QUESTIONS DE RÉVISION

1. Nommez les trois parties fondamentales d'un additionneur DCB.
2. Décrivez le circuit qui détecte le besoin d'une correction dans un additionneur DCB et dites comment cette correction est appliquée.

6-17 MULTIPLICATEURS BINAIRES

La multiplication de deux nombres binaires faite manuellement se résume à des additions successives et à des décalages. Voici une illustration:

```
   1011     multiplicande (11)
 × 1101     multiplicateur (13)
  1011
 0000
1011
1011
10001111    produit (143)
```

Cette façon de procéder est identique à l'examen des bits successifs du multiplicateur, en commençant par le bit de poids faible. Si ce bit est 1, on recopie le multiplicande. S'il est 0, on inscrit des 0. Par rapport à la ligne précédente, les nombres qui sont écrits sur les lignes suivantes sont décalés d'un rang vers la gauche. Lorsque tous les bits du multiplicateur ont été examinés, on les additionne pour obtenir le produit.

Dans les machines numériques, cette façon de procéder est quelque peu modifiée du fait que les additionneurs binaires ne peuvent traiter que deux nombres à la fois. Ainsi, au lieu d'additionner toutes les lignes jusqu'à la fin, on les additionne deux par deux et on inscrit les sommes intermédiaires dans un registre (l'accumulateur). En outre, quand le bit du multiplicateur est 0, il n'est pas nécessaire d'écrire les 0 et de les additionner, vu

qu'ils ne changent rien au résultat définitif. Refaisons la multiplication précédente en suivant un processus quelque peu différent.

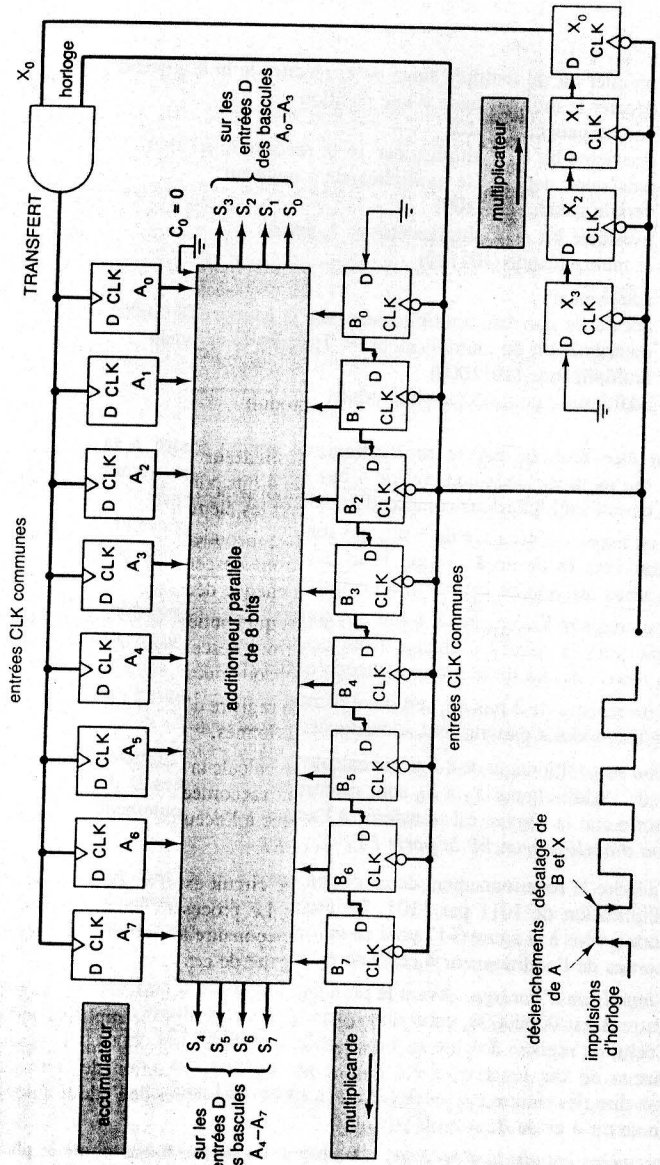
```
multiplicande   1011
multiplicateur  1101
      1011   premier bit du multiplicateur = 1; réécrire le multiplicande;
             décaler le multiplicande d'une position
             vers la gauche (10110)
      1011   deuxième bit du multiplicateur = 0; réécrire le résultat
             précédent; déplacer le multiplicande à nouveau
             vers la gauche (101100)
+ 101100     troisième bit du multiplicateur = 1; écrire
             le multiplicande (101100)
------
    110111   additionner
             décaler le nouveau multiplicande vers la gauche (1011000)
+ 1011000    quatrième bit du multiplicateur = 1; écrire le nouveau
             multiplicande (1011000)
  10001111   additionner pour obtenir le produit
```

La multiplication peut être réalisée par le **multiplicateur série** illustré à la figure 6-16 qui multiplie deux nombres de 4 bits pour donner un produit ayant 8 bits de longueur. Le circuit multiplicateur comprend les éléments que voici:

- *Un registre X.* Soit un registre à décalage de 4 bits qui contient le multiplicateur. Il décale son contenu vers la droite à chaque front descendant de l'horloge. Remarquez que des zéros arrivent de la gauche à chaque décalage.
- *Un registre B.* Soit un registre à décalage de 8 bits qui contient le multiplicande. Il décale son contenu vers la gauche à chaque front descendant de l'horloge. Remarquez que des zéros arrivent de la droite à chaque décalage.
- *Un registre A.* Soit un registre de 8 bits qui sert de registre d'accumulateur; c'est dans ce registre que les produits partiels sont additionnés.
- *Un additionneur.* Soit un additionneur de 8 bits qui calcule la somme des registres A et B. Les sorties de l'additionneur S_7 à S_0 sont raccordées aux entrées D de l'accumulateur de sorte que la somme est transférée à l'accumulateur seulement *quand une impulsion d'horloge franchit la porte ET.*

La meilleure façon de décrire le fonctionnement de ce circuit est d'y aller étape par étape pour le cas de la multiplication de 1011 par 1101. Le processus entier demande quatre cycles d'horloge. Reportez-vous à la figure 6-17 pour connaître le contenu de chaque registre et la valeur sur les sorties de l'additionneur à chacune de ces étapes.

1. *Avant la première impulsion d'horloge.* Avant la première impulsion d'horloge, le contenu du registre A est 00000000, celui du registre B est le multiplicande soit 00001011, et celui du registre X est le multiplicateur, soit 00001101. Nous supposons que chacun de ces registres a été rempli par l'action des entrées asynchrones (c'est-à-dire, les entrées RAU et RAZ). Les sorties de l'additionneur contiennent la somme de A et de B, soit 00001011.
2. *À l'instant de la première impulsion d'horloge.* Comme le bit de poids le plus faible du multiplicateur (X_0) est 1, la première impulsion franchit la porte ET, et son FM transfère les sorties somme dans l'accumulateur. Le FD qui suit provoque le décalage vers la droite et la gauche respectivement des registres X et B. Ceci, évidemment, génère une nouvelle somme de A et B.



Le registre A réagit au passage du front *montant* du signal d'horloge les registres B et X réagissent au passage du front *descendant*.

FIGURE 6-16 Circuit d'un multiplicateur série 4 bits dans lequel le multiplicande et le multiplicateur sont dans des registres à décalage.

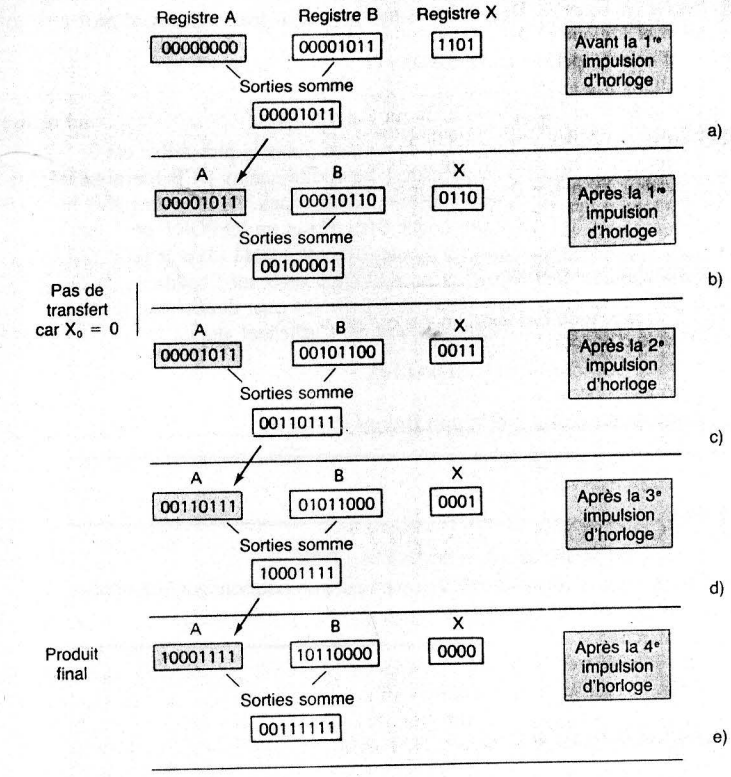


FIGURE 6-17 Contenu de différents registres pendant les étapes de la multiplication de 1011 par 1101.

3. *À l'instant de la deuxième impulsion d'horloge.* Le second bit du multiplicateur initial se trouve désormais dans la position X_0. Comme il s'agit d'un zéro, la seconde impulsion d'horloge est bloquée et n'atteint pas l'accumulateur. Donc, le chiffre formé par les sorties somme n'est pas transféré dans l'accumulateur, et le nombre dans l'accumulateur reste inchangé. Le FD de l'impulsion d'horloge, une nouvelle fois, décale les registres X et B.
4. *À l'instant de la troisième impulsion d'horloge.* Le troisième bit du multiplicateur initial est maintenant dans la position X_0. Comme il s'agit d'un 1, le FM de l'horloge initie le transfert de la somme dans le registre A. Le FD décale les registres X et B, à nouveau.
5. *À l'instant de la quatrième impulsion d'horloge.* Le dernier bit du multiplicateur initial est maintenant dans la position X_0. Comme il s'agit d'un 1, le FM du quatrième signal d'horloge transfère la somme dans l'accumulateur. Ce dernier contient désormais le produit final. Le FD décale une dernière fois les registres A et X. Remarquez que le contenu de X est 0000, étant donné que tous les chiffres du multiplicateur ont été sortis.

Multiplication logicielle De nombreux microprocesseurs ne disposent pas dans leur unité arithmétique et logique (UAL) des circuits (*matériel*) en mesure d'effectuer la multiplication, ni d'autres opérations de calcul plus complexes (par exemple, division, racine carrée, fonctions trigonométriques). Les micro-ordinateurs basés sur de tels microprocesseurs accomplissent la multiplication au moyen d'un programme (*logiciel*) qui réalise une multiplication en répétant des instructions d'addition et de décalage. D'autres programmes de même nature existent pour effectuer d'autres opérations arithmétiques. Le recours à un logiciel se traduit par moins de circuits dans le microprocesseur, mais aussi, dans la plupart des cas, par un temps de calcul plus long que la contrepartie matérielle.

QUESTIONS DE RÉVISION

1. Décrivez la séquence des événements qui se déroulent dans le circuit multiplicateur pendant 1 cycle d'horloge.
2. *Vrai ou faux:* Les sorties de l'additionneur sont transférées dans l'accumulateur à *chaque* FM du signal d'horloge.

6-18 FONCTIONS DE CALCUL COMPLEXES EN CI

Les familles de la technologie TTL et CMOS contiennent plusieurs circuits intégrés qui effectuent des calculs arithmétiques complexes. Décrivons quelques-uns des éléments existants sur le marché.

Les CI 74LS381 et 74HC381 sont appelés *générateurs de fonctions/unités logiques-arithmétiques*. Ils réalisent huit opérations arithmétiques et logiques diverses sur des entrées de 4 bits. Parmi ces opérations, sélectionnées au moyen de trois entrées SÉLECTION, mentionnons: l'addition, la soustraction, l'addition logique (OU), le OU exclusif et la multiplication logique (ET).

Les 74284 et 74285 sont des *multiplicateurs binaires parallèles*. À l'instar du multiplicateur série de la figure 6-16, ces circuits intégrés renferment les circuits en mesure de générer le produit de deux nombres binaires de 4 bits. À l'opposé du multiplicateur série, ces CI accomplissent une multiplication parallèle en se servant seulement de circuits à logique combinatoire, circuits qui inspectent les données d'entrée de 4 bits et génèrent le produit de 8 bits. Le multiplicateur série est au moins 4 fois plus lent, puisqu'il doit décaler les registres du multiplicande et du multiplicateur 4 fois.

Le AM9511 est un élément LSI appelé *unité de traitement arithmétique*. C'est un boîtier de 24 broches qui effectue nombre d'opérations de calcul complexes, comme *l'addition, la soustraction, la multiplication, la division, l'extraction de la racine carrée, des fonctions trigonométriques* (par exemple, *sin*, *cos*, *tan*) *et la transformation logarithmique*. Cette unité réalise ces opérations sur des nombres d'entrée de 16 bits ou de 32 bits. Elle a recours à un signal d'horloge (comme le multiplicateur série de la figure 6-16) pour ordonnancer ses opérations. Plus le calcul est complexe, plus il faut compter de cycles d'horloge.

L'unité de traitement arithmétique a été conçue pour recevoir ses données et ses commandes du microprocesseur. Une fois munie de ses données et de ses commandes, l'unité effectue l'opération indiquée dans le nombre de cycles d'horloge requis. Puis, elle signifie au microprocesseur la fin du traitement. Ce dernier se charge généralement de recueillir sur les broches de l'unité les données de sortie (le résultat) et de les placer dans ses registres internes en vue du traitement suivant.

6-19 SYMBOLES IEEE/ANSI

La figure 6-18 a) nous fait voir le symbole IEEE/ANSI d'un additionneur complet d'un bit. Notez la lettre Σ utilisée pour désigner l'opération d'addition. La figure 6-18 b) représente le symbole IEEE/ANSI pour un additionneur parallèle 4 bits comme le 7483 ou le 74283. Observez aussi comment les lettres P et Q sont employées pour représenter les entrées de 4 bits; le signe Σ servant toujours à indiquer la somme fournie en sortie. Les lettres P, Q et Σ sont prescrites dans la norme IEEE/ANSI et doivent figurer à l'intérieur du périmètre du symbole. Les lettres employées pour identifier les entrées et les sorties externes au symbole ne sont pas spécifiées, et nous sommes justifié d'utiliser A, B et S comme dans nos discussions précédentes.

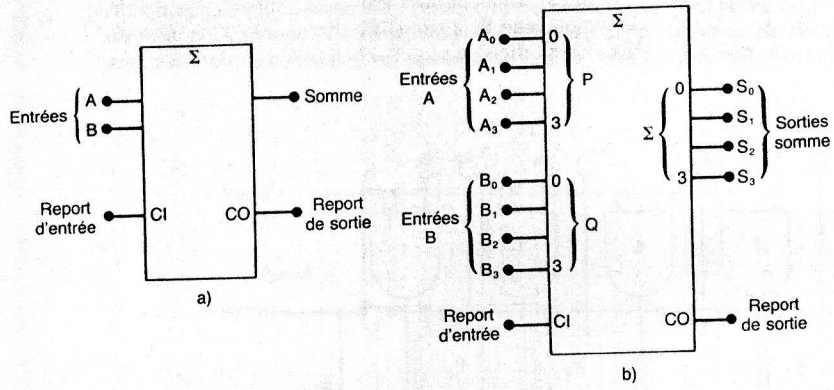


FIGURE 6-18 Symboles IEEE/ANSI a) d'un additionneur complet 1 bit; b) d'un CI additionneur parallèle de 4 bits (7483/74283).

6-20 ÉTUDE D'UN CAS DE DÉPANNAGE

Un technicien se voit confier la vérification du circuit additionneur/soustracteur de la figure 6-13; durant son test, il relève les observations suivantes pour les différents modes:

- **Mode 1: ADD = 0, SUB = 0.** Les sorties de la somme contiennent toujours un nombre égal à la valeur dans le registre A *plus un*. Par exemple, quand $[A] = 0110$, la somme, elle, est $[S] = 0111$. C'est une réponse erronée, puisque les sorties des portes OU et C_0 doivent être 0 dans ce mode pour que $[S] = [A]$.
- **Mode 2: ADD = 1, SUB = 0.** La somme excède toujours de 1 la bonne réponse. Par exemple, avec $[A] = 0010$ et $[B] = 0100$, la somme en sortie est 0111 et non 0110.
- **Mode 3: ADD = 0, SUB = 1.** Les sorties S sont toujours égales à $[A] - [B]$, comme il se doit.

Quand il examine ses résultats d'essai, le technicien constate que les sorties somme dépassent la somme prévue de 1 dans les deux premiers modes de fonctionnement. À prime abord, il soupçonne une défaillance possible de l'une des entrées de rang faible de l'additionneur, mais rejette cette hypothèse, car une panne à ce niveau toucherait également

la soustraction qui fonctionne comme prévu. Il finit par réaliser qu'il existe bien un dérèglement qui ajoute un 1 supplémentaire aux résultats dans les deux premiers modes sans causer d'erreur dans le mode de soustraction.

Rappelons que C_0 est forcé à 1 en mode de soustraction dans le cadre de la complémentation à 2 de $[B]$. Pour les autres modes, C_0 est à 0. Le technicien vérifie la connexion entre le signal SUB et l'entrée C_0 de l'additionneur; il relève un circuit ouvert imputable à une mauvaise soudure. Cette connexion ouverte explique le 1 supplémentaire trouvé dans les résultats en modes 1 et 2. Cette connexion n'a aucun effet sur la soustraction, étant donné que C_0 doit toujours être 1 dans ce cas.

EXEMPLE 6-12

Considérez de nouveau le circuit additionneur/soustracteur. Supposez qu'il y ait une rupture dans le chemin de connexion entre l'entrée SUB et les portes ET, comme celle illustrée sur la figure 6-19. Décrivez les effets de ce circuit ouvert sur le fonctionnement dans tous les modes.

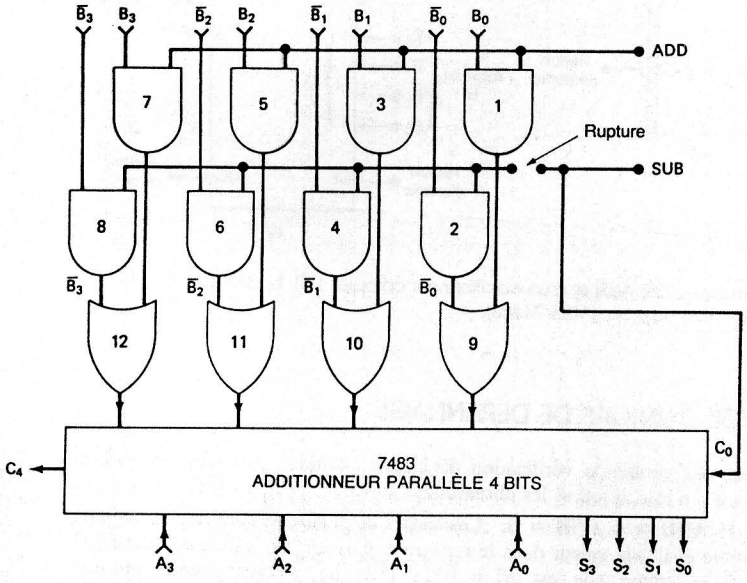


FIGURE 6-19 Exemple 6-12

Solution D'abord, il faut réaliser que cette rupture a pour effet de produire un 1 logique sur les entrées correspondantes des portes ET 2, 4, 6 et 8, ce qui valide en permanence ces portes qui laissent alors passer les valeurs de $[B]$ jusqu'aux portes OU.

Mode 1: ADD = 0, SUB = 0. La rupture en question transforme le circuit en un circuit quasi-soustracteur. Le complément à 1 de $[B]$ atteint les sorties des portes OU et est additionné à $[A]$ dans l'additionneur. Avec $C_0 = 0$, le complément à 2 de $[B]$ n'est pas complet, il lui manque l'addition du 1 sur le rang le plus faible.

Donc, l'additionneur effectue l'opération $[A] - [B] - 1$. Pour démontrer cela, travaillons l'exemple suivant: $[A] = +6 = 0110$ et $[B] = +3 = 0011$. L'additionneur produit ceci:

le complément à 1 de $[B]$ = 1100
$[A]$ = 0110
résultat = 10010
↑ ce report est rejeté

Le résultat est 0010, soit +2, au lieu de la bonne réponse 011 = +3.

Mode 2: ADD = 1, SUB = 0. Quand ADD = 1, les portes ET 1, 3, 5 et 7 laissent passer les entrées de B à travers les portes OU de l'étape suivante. Donc, chaque porte OU retrouve sur ses entrées $\overline{B}$ et B, ce qui donne toujours 1 en sortie. Par exemple, les entrées de la porte ET 9 sont $\overline{B}_0$ provenant de la porte ET 2 (attribuable à la défaillance) et B_0 provenant de la porte ET 1 (puisque ADD = 1). Donc la porte OU 9 donne en sortie $\overline{B}_0 + B_0 = 1$.

L'additionneur ajoutera le nombre 1111 formé par les sorties des portes OU à $[A]$ pour donner une somme qui est la valeur de $[A]$ moins 1. Pourquoi? Parce que $1111_2 = -1_{10}$.

Mode 3: ADD = 0, SUB = 1. Ce mode fonctionne correctement, puisque SUB = 1 débloque de toute façon les portes ET 2, 4, 6 et 8.

PROBLÈMES

SECTION 6-1

6-1 Additionnez les groupes de nombres binaires que voici en suivant les règles de l'addition binaire:
a) 1010 + 1011 b) 1111 + 0011 c) 1011,1101 + 11,1
d) 0,1011 + 0,1111 e) 10011011 + 10011101

SECTION 6-2

6-2 Représentez chacun des nombres décimaux signés que voici selon la notation en complément à 2. Utilisez un total de 8 bits y compris le bit de signe.
a) +32 b) −14 c) +63 d) −104
e) −1 f) −128 g) +169 h) 0

6-3 Voici le complément à 2 de nombres décimaux signés. Trouvez la valeur décimale correspondante.
a) 01101 b) 11101 c) 01111011 d) 10011001 e) 01111111
f) 100000 g) 11111111 h) 10000001

6-4 a) Dites quelle est la gamme des valeurs décimales signées que l'on peut représenter avec 12 bits y compris le bit de signe.
b) Dites combien il faut de bits pour représenter les nombres décimaux compris entre −32768 et +32767.

6-5 Énumérez dans l'ordre tous les nombres signés qu'il est possible de représenter avec 5 bits au moyen de la notation en complément à 2.

6-6 Un ordinateur a mémorisé le nombre binaire suivant: 10100100. Seul le programmeur qui a placé le groupe de données en mémoire en connaît la signification. Voici une liste de possibilités:

a) Un nombre décimal non signé
b) Un nombre décimal signé exprimé dans la notation en complément à 2
c) Un nombre décimal représenté en DCB
d) Un code ASCII associé à un bit de parité
e) Un nombre décimal signé exprimé dans la notation signe-grandeur.

Pour chacune de ces possibilités, trouvez la signification du nombre 10100100.

6-7 Quel intervalle des valeurs décimales non signées peut-on représenter avec 10 bits? Quel intervalle des valeurs décimales signées peut-on représenter avec 10 bits?

SECTIONS 6-3 et 6-4

6-8 Vous comprendrez immédiatement pourquoi la notation signe-grandeur n'est pas utilisée dans les ordinateurs pour représenter des nombres signés si vous effectuez les trois calculs suivants:

a) Représentez +12 au moyen de 5 bits dans la notation signe-grandeur.
b) Représentez −12 au moyen de 5 bits dans la notation signe-grandeur.
c) Additionnez les deux nombres binaires trouvés et constatez que la somme ne ressemble pas du tout à zéro.

6-9 Effectuez les opérations suivantes avec la notation en complément à 2. Utilisez pour chaque nombre 8 bits (y compris le bit de signe). Contrôlez vos résultats en convertissant la réponse binaire en décimale.

a) additionner +9 à +6
b) additionner +14 à −17
c) additionner +19 à −24
d) additionner −48 à −80
e) soustraire +16 de +17
f) soustraire +21 de −13
g) soustraire +47 de +47
h) soustraire −36 de −15
i) additionner +17 à −17
j) soustraire −17 de −17

6-10 Même question que le problème 6-9 pour les cas suivants; montrez qu'il y a dans chaque cas dépassement.

a) additionner +37 à +95
b) soustraire +37 de −95

SECTIONS 6-5 et 6-6

6-11 Multipliez les paires de nombres binaires suivants:

a) 111 × 101
b) 1011 × 1011
c) 101,101 × 110,010
d) ,1101 × ,1011

6-12 Effectuez les divisions suivantes:

a) 1100 ÷ 100
b) 111111 ÷ 1001
c) 10111 ÷ 100
d) 10110,1101 ÷ 1,1

SECTIONS 6-7 et 6-8

6-13 Additionnez les nombres décimaux suivants après les avoir convertis dans le code DCB:

a) 74 + 23
b) 58 + 37
b) 147 + 380
d) 385 + 118

6-14 Trouvez la somme de chacune des paires suivantes de nombres hexadécimaux:

a) 3E91 + 2F93
b) 91B + 6F2
c) ABC + DEF

6-15 Soustrayez les paires de nombres hexadécimaux suivantes:

a) 3E91 − 2F93
b) 91B − 6F2
c) 0300 − 005A

6-16 Le manuel d'utilisateur d'un ordinateur personnel indique que les places mémoires utilisables de ce dernier ont les adresses hexadécimales suivantes: 0200 à 03FF et 4000 à 7FD0. Dites quel est le nombre total de places mémoires disponibles.

SECTION 6-11

6-17 Transformez le circuit AC de la figure 6-7 en un circuit n'ayant que des portes NON-ET.

6-18 Dressez la table de vérité d'un demi-additionneur (entrées *A* et *B*, sorties SOMME et REPORT). À partir de cette table de vérité, concevez un circuit logique qui agirait comme un demi-additionneur.

6-19 Un additionneur complet peut être réalisé de différentes façons. La figure 6-20 en montre une qui utilise deux demi-additionneurs. Construisez la table logique correspondant à ce montage et vérifiez qu'elle correspond à un additionneur complet.

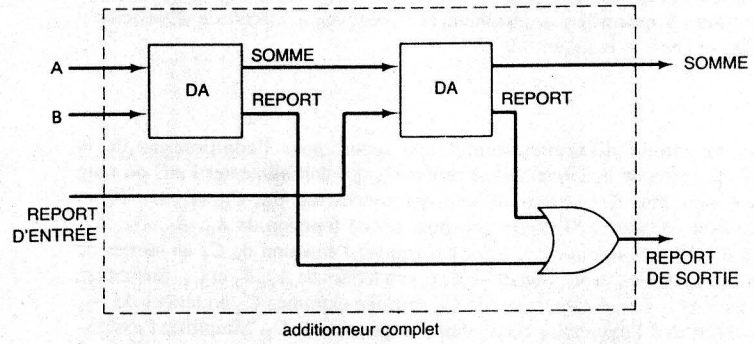


FIGURE 6-20

6-20 Reportez-vous à la figure 6-9. Déterminez le contenu du registre *A* après la suite d'opérations que voici: [*A*] = 000, [0101] ⟶ [*B*], [*S*] ⟶ [*A*], [1011] ⟶ [*B*], [*S*] ⟶ [*A*].

6-21 Reportez-vous à la figure 6-9. Supposez que chaque bascule a $t_{PLH} = t_{PHL} =$ 30 ns, que le temps de stabilisation est 10 ns et que chaque AC a un retard de propagation de 40 ns. Dites quelle est la durée minimale admissible entre le FM de l'impulsion CHARGE et le FM de l'impulsion TRANSFERT pour que l'additionneur fonctionne correctement.

6-22 Dans les circuits d'additionneur et de soustracteur examinés dans ce chapitre, nous n'avons pas pris en considération la possibilité d'un *dépassement*. Il y a dépassement quand les deux nombres additionnés ou soustraits donnent un résultat qui contient plus de bits que la capacité de l'accumulateur. Par exemple, il est possible de ranger dans des registres de 4 bits, y compris le bit de signe, des nombres allant de +7 à −8 (en complément à 2). Donc, si le résultat d'une addition ou d'une soustraction

ne se trouve pas dans la gamme de $+7$ à -8, nous disons qu'il y a dépassement. En cas de dépassement, les résultats sont sans valeur puisqu'il ne sont pas mémorisés correctement dans l'accumulateur. À titre d'illustration, additionnez $+5$ (0101) et $+4$ (0100), dont le résultat est 1001. L'interprétation du nombre 1001 est fausse, car elle indique un nombre négatif puisqu'il y a un 1 comme bit de signe.

Dans les ordinateurs, il existe habituellement des circuits qui détectent une condition de dépassement. Il existe plusieurs solutions pour détecter le dépassement. Une des méthodes utilisables dans le cas de l'additionneur traitant des nombres exprimés en complément à 2 est la suivante:

1. Examinez les bits de signe des deux nombres à additionner.
2. Examinez le bit de signe du résultat.
3. Il y a dépassement quand les deux nombres additionnés sont *tous les deux positifs* et que le bit de signe du résultat est 1 ou quand les deux nombres sont *tous les deux négatifs* et que le bit de signe du résultat est 0.

On peut vérifier que cette méthode est exacte en essayant quelques exemples.

Faites les exemples suivants: 1) $5 + 4$; 2) $-4 + (-6)$; 3) $3 + 2$. Les cas 1 et 2 donnent lieu à un dépassement tandis que le cas 3 n'en produit pas. Donc en examinant les bits de signe on peut construire un circuit logique pour générer une valeur de sortie 1 quand il y a dépassement. Concevez le circuit de dépassement pour l'additionneur de la figure 6-9.

SECTION 6-13

6-23 Concevez un circuit d'examen anticipé du report pour l'additionneur de la figure 6-9; la sortie de ce circuit est le report C_3 qui doit alimenter l'AC du rang de poids le plus fort. Cette sortie dépend des entrées A_0, B_0, C_0, A_1, B_1, A_2 et B_2. Autrement dit, dérivez l'expression pour C_3 en fonction de A_0, B_0, C_0, A_1, B_1, A_2 et B_2. (*Suggestion*: commencez par trouver l'équation de C_1 en termes de A_0, B_0 et C_0. Ensuite trouvez l'équation de C_2 en termes de A_1, B_1 et C_1. Substituez l'expression de C_1 dans l'expression de C_2. Ensuite exprimez C_3 en termes de A_2, B_2 et C_2. Substituez l'expression de C_2 dans l'expression de C_3. Simplifiez l'expression définitive de C_3 et exprimez-la sous la forme d'une somme de produits. Construisez le circuit.)

SECTION 6-14

6-24 Illustrez toutes les valeurs logiques intermédiaires sur les entrées et les sorties de la figure 6-10 a) pour l'addition de 354_8 à 103_8.

SECTION 6-15

6-25 Pour le circuit de la figure 6-13, déterminez les sorties de la somme dans les cas suivants:

a) registre $A = 0101$ $(+5)$, registre $B = 1110$ (-2); SUB $= 1$, ADD $= 0$
b) registre $A = 1100$ (-4), registre $B = 1110$ (-2); SUB $= 0$, ADD $= 1$

6-26 Modifiez le circuit de la figure 6-13 afin de n'avoir qu'une entrée de commande, X, au lieu des deux entrées ADD et SUB. Ce circuit doit être un additionneur quand $X = 0$ et un soustracteur quand $X = 1$. Ensuite simplifiez chaque groupe de portes. (*Suggestion*: Remarquez que maintenant chaque groupe de portes fonctionne comme un inverseur commandé.)

SECTION 6-16

6-27 Supposez que les valeurs suivantes se trouvent sur les entrées de la figure 6-14: $[A] = 0101$, $[B] = 1001$, $C_0 = 0$. Déterminez les niveaux logiques de $[S]$, X, $[\Sigma]$ et REPORT.

6-28 Dites si l'additionneur DCB de la figure 6-14 fonctionnera autrement si C_0 de l'additionneur du haut est gardé à 0, pendant que C_0 de l'additionneur du bas devient le report d'entrée. Expliquez.

6-29 Supposez que le registre A de la figure 6-15 contient la représentation codée DCB de 376 et que le registre B contient la représentation DCB de 469. Déterminez les valeurs sur ces sorties.

SECTION 6-17

6-30 Reportez-vous au circuit de la figure 6-16. Supposez qu'au début le registre B contient le multiplicande 0111 et que le registre X contient le multiplicateur 1001. En outre, supposez que $[A] = 00000000$ avant la première impulsion d'horloge. Déterminez les contenus des registres A, B et X après chaque impulsion d'horloge.

SECTION 6-20

6-31 Le circuit multiplicateur série de la figure 6-16 est testé pour différents couples de multiplicande et de multiplicateur. Voici le tableau des résultats trouvés:

Multiplicande	Multiplicateur	Résultat
4	5	60
3	4	45
15	2	225
2	15	30
1	7	15

L'examen de ces résultats montre que le fonctionnement de ce circuit est défectueux. Analysez chacune des défaillances possibles suivantes, et pour chacune indiquez si oui ou non, elle peut être la cause du dérèglement observé. Expliquez chacune de vos réponses. (*Suggestion:* Essayez de dégager une constante dans ces résultats.)

a) Les entrées de la porte ET sont court-circuitées.
b) Il y a une rupture dans la connexion X_0 de la porte ET.
c) L'entrée D de la bascule X_3 est ouverte.

6-32 Supposez qu'il y ait une rupture dans la connexion aboutissant à C_0 du multiplicateur série de la figure 6-16. Montrez quelle modification au fonctionnement de ce circuit introduit une telle panne.

6-33 Le multiplicateur série de la figure 6-16 est testé par un technicien qui relève les observations suivantes:

Multiplicande	Multiplicateur	Résultat
2	3	6
3	4	12
4	6	8
6	4	8
9	3	11
12	8	0
7	7	1

Examinez les données consignées dans ce tableau et tentez de dégager une ou plusieurs causes possibles à ce dérèglement.

6-34 L'additionneur DCB de la figure 6-14 est testé et on relève les observations suivantes:

	$B_1B_2B_3B_4$	$A_3A_2A_1A_0$	$\Sigma_3\Sigma_2\Sigma_1\Sigma_0$	REPORT (X)
(1)	0011	0110	1001	0
(2)	0111	1000	1111	0
(3)	1001	1001	0010	0

Envisagez les défaillances suivantes et indiquez si oui ou non elles peuvent causer le dérèglement observé. Expliquez chacune de vos réponses.

a) Les entrées A_1 et A_0 de l'additionneur de la correction sont court-circuitées par l'intérieur.
b) Il y a une rupture de la connexion entre X et l'additionneur de la correction.
c) Les entrées de la porte OU supérieure sont court-circuitées par l'intérieur.
d) La sortie de la porte ET est « gelée » au niveau BAS.

CONTRÔLE

6-35 Définissez chacun des termes suivants:

a) Additionneur complet
b) Représentation en complément à 2
c) Unité arithmétique et logique
d) Bit de signe
e) Dépassement
f) Accumulateur
g) Additionneur parallèle
h) Report d'anticipation

APPLICATION AU MICRO-ORDINATEUR

6-36 Dans l'unité arithmétique et logique d'un microprocesseur type, les résultats d'une opération arithmétique sont généralement (mais pas toujours) placés dans le registre accumulateur, comme dans les figures 6-9, 6-13 et 6-16. Également, dans la plupart des UAL, le résultat de chaque calcul arithmétique contrôle les états de certaines bascules spéciales appelées *drapeaux ou indicateurs*. Ces drapeaux sont examinés par le microprocesseur qui, à partir de leur valeur, prend des décisions durant l'exécution de certains types d'instructions. Les drapeaux les plus utilisés sont:

S (drapeau de signe). Cette bascule a toujours comme valeur de sortie la valeur du signe de la dernière réponse obtenue par l'UAL.

Z (drapeau zéro). Ce drapeau est mis à 1 quand le résultat d'une opération de l'UAL est précisément zéro. Autrement la valeur de ce drapeau est 0.

C (drapeau de report). Cette bascule a toujours la même valeur que le report du bit de rang le plus fort de l'UAL.

En considérant que le circuit additionneur/soustracteur de la figure 6-13 est votre unité arithmétique et logique, concevez le circuit logique qui matérialise ces trois drapeaux. Les sorties somme et C_4 doivent servir au positionnement de la valeur dans chaque drapeau au moment de l'arrivée de l'impulsion de TRANSFERT. Par exemple, si la somme est exactement zéro (soit 0000), le drapeau Z doit être mis à 1 à l'instant du FM de TRANSFERT, autrement il doit être positionné à zéro.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 6-1

1. a) 11101 **b)** 101,111 **c)** 10010000

SECTION 6-2

1. a) 01101 **b)** 11001 **c)** 10000
2. a) −29 **b)** −64 **c)** +63
3. −2048 à +2047 **4.** Sept **5.** −32768
6. 10000, 10000000, 1000

SECTION 6-3

1. Vrai **2. a)** $100010_2 = -30_{10}$
b) $000000_2 = 0_{10}$

SECTION 6-4

1. a) $01111_2 = +15_{10}$; **b)** $11111_2 = -1_{10}$.
2. En comparant le signe de bit de la somme avec les signes de bit des nombres additionnés.

SECTION 6-5

1. 1100010

SECTION 6-7

1. La somme d'au moins un rang décimal dépasse 1001 (9). **2.** Le facteur de correction est ajouté au rang des unités et à celui des dizaines.

SECTION 6-8

1. 923 **2.** 3DB

SECTION 6-10

1. Trois; deux **2 a)** $S_2 = 0, C_3 = 1$
b) $C_5 = 0$

SECTION 6-12

1. Un, quatre, quatre **2.** 0100

SECTION 6-15

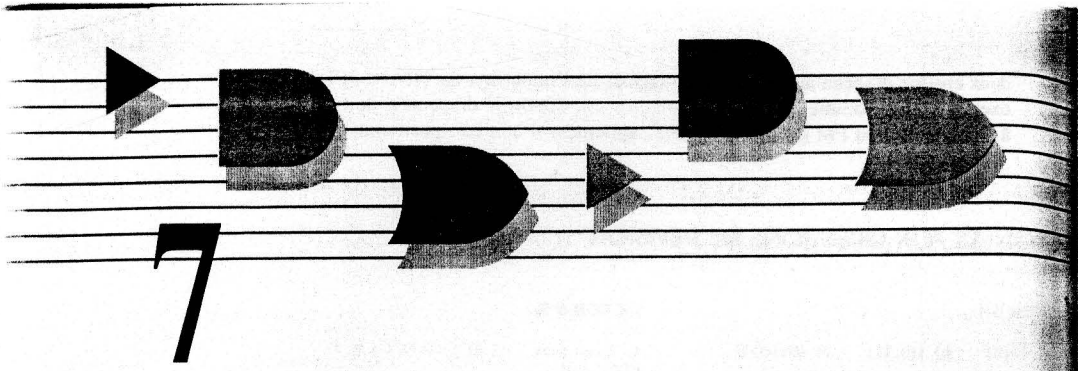
1. Pour ajouter le 1 nécessaire pour compléter la complémentation à 2 du nombre dans le registre B.
2. 0010 **3.** 1101 **4.** Faux; le complément à 1 apparaît ici.

SECTION 6-16

1. Deux additionneurs de 4 bits et les circuits logiques de correction. **2.** Les circuits logiques de correction décèlent une somme supérieure à 9 et commande l'addition de 0110 à cette somme.

SECTION 6-17

1. La somme courante est transférée dans A; le multiplicande et le multiplicateur sont décalés; l'additionneur calcule une nouvelle somme.
2. Faux; seulement quand le bit du multiplicateur (X_0) est HAUT.

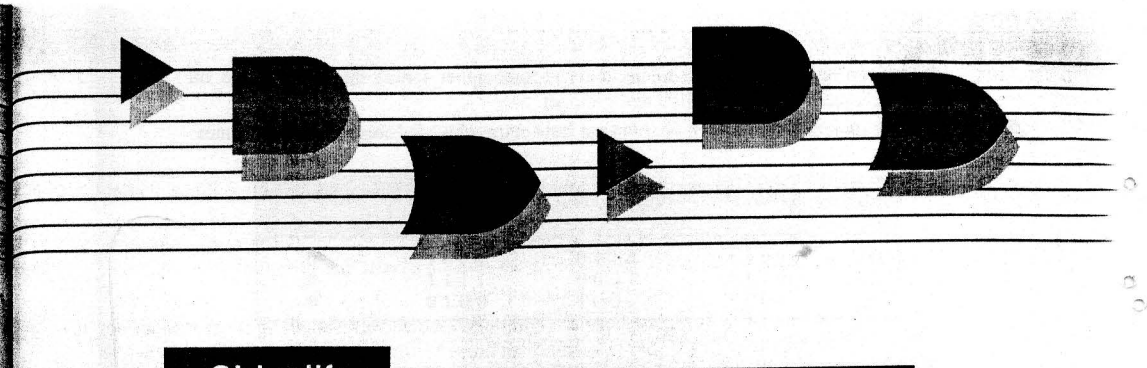


7

COMPTEURS ET REGISTRES

Contenu

7-1 Compteurs asynchrones (à propagation)
7-2 Compteurs à MODULO < 2^N
7-3 Compteurs asynchrones CI
7-4 Décompteurs asynchrones
7-5 Retard de propagation dans les compteurs asynchrones
7-6 Compteurs synchrones (parallèles)
7-7 Décompteurs et compteurs réversibles parallèles
7-8 Compteurs préréglables
7-9 Le compteur 74193 (LS193/HC193)
7-10 Complément à la notation fonctionnelle IEEE/ANSI
7-11 Décodage d'un compteur
7-12 Décodage des signaux transitoires
• 7-13 Compteurs DCB montés en cascade
7-14 Conception d'un compteur synchrone
7-15 Compteurs circulaires
7-16 Applications: compteur de fréquence
7-17 Applications: horloge numérique
7-18 Registres intégrés
7-19 Écriture parallèle/lecture parallèle — les puces 74174 et 74178
7-20 Écriture série/lecture série — la puce 4731B
7-21 Écriture parallèle/lecture série — la puce 74165/74LS165/74HC165
7-22 Écriture série/lecture parallèle — la puce 74164/74LC164/74HC164
7-23 Symboles IEEE/ANSI pour les registres



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Comprendre le fonctionnement et les caractéristiques des compteurs asynchrones et parallèles.
- Construire des compteurs ayant un MODULO inférieur à 2^N.
- Reconnaître les symboles IEEE/ANSI utilisés dans les registres et compteurs intégrés.
- Construire des compteurs et des décompteurs.
- Analyser et évaluer les différents types de compteurs préréglables.
- Concevoir des compteurs synchrones à séquence de nombres quelconque.
- Comprendre plusieurs types de solutions pour le décodage des divers compteurs.
- Éliminer les transitoires de décodage en recourant à la technique appelée strobage.
- Comparer les principales différences entre les compteurs à anneau et les compteurs Johnson.
- Analyser le principe de fonctionnement d'un fréquencemètre et d'une horloge numérique.
- Reconnaître les divers types de registre et en expliquer le fonctionnement.
- Appliquer les techniques de dépannage exposées pour les circuits combinatoires au dépannage de systèmes logiques séquentiels.

Introduction

Au chapitre 5, nous avons vu qu'il est possible de raccorder des bascules pour qu'elles fonctionnent comme des compteurs ou des registres. Nous avons étudié à ce moment-là des compteurs et des registres élémentaires. Les systèmes numériques font appel à plusieurs variantes de ces circuits de base, particulièrement sous forme de circuits intégrés. Le but du présent chapitre est de vous montrer comment on parvient à construire différents types de compteurs et de registres en combinant des bascules et des portes logiques.

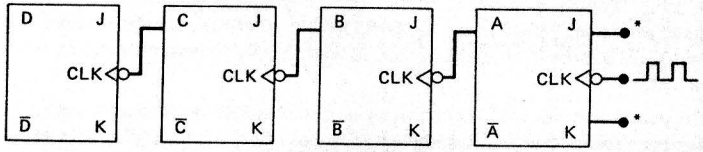
Vous serez initié à plusieurs applications qui vous illustreront diverses manières de mettre en oeuvre les circuits logiques dans les systèmes numériques. Nous examinerons aussi quelques-uns des compteurs et registres intégrés.

7-1 COMPTEURS ASYNCHRONES (À PROPAGATION)

La figure 7-1 reproduit le compteur de 4 bits déjà étudié au chapitre 5. Rappelons les points suivants concernant son fonctionnement:

1. Les impulsions d'horloge sont appliquées à la borne *CLK* de la bascule *A*. Celle-ci commute (passe dans l'état opposé) chaque fois qu'arrive un front descendant (transition niveau HAUT à niveau BAS). Notez que dans toutes ces bascules $J = K = 1$.
2. La sortie normale de la bascule *A* fait office de signal d'horloge pour la bascule *B*, de sorte que cette dernière commute chaque fois que le signal provenant de *A* passe de 1 à 0. De la même manière, *C* commute quand la sortie *B* passe de 1 à 0, et *D* commute quand *C* passe de 1 à 0.
3. Le tableau de la figure 7-1 montre la suite des états binaires pris par les bascules après chaque impulsion d'horloge. Si on imagine que les sorties des bascules *D*, *C*, *B* et *A* représentent un nombre binaire, *D* étant le bit de poids le plus fort et *A* le bit de poids le plus faible, la suite des nombres produite par ces bascules va de 0000 à 1111.

FIGURE 7-1 Compteur asynchrone (à propagation) de quatre bits.



* toutes les entrées J et K sont supposées à 1.

D	C	B	A	Nombre d'impulsions
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15
0	0	0	0	16 recyclage
0	0	0	1	17
0	0	1	0	18
0	0	1	1	19
0	1	0	0	20
0	1	0	1	21
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

4. Après la 15e implusion d'horloge, les bascules du compteur sont dans la condition 1111. Quand la 16e impulsion atteint la borne *CLK* de la bascule *A*, la sortie *A* passe de 1 à 0, ce qui fait commuter la bascule *B* dont la sortie passe aussi de 1 à 0, et ainsi de suite jusqu'à ce que le compteur affiche 0000. Autrement dit, le compteur a parcouru un cycle entier (0000 à 1111) et est revenu à son état initial; on dit que le compteur est *recyclé* et qu'il recommence le cycle de dénombrement des impulsions qui arrivent.

Ce type de compteur, dans lequel la sortie de chaque bascule agit comme le signal d'horloge de la suivante, est appelé un **compteur asynchrone**. Ce nom vient du fait que les quatre bascules *ne* changent *pas* d'état toutes en même temps à la transition du signal d'horloge. Seule la bascule *A* commute quand arrivent les impulsions d'horloge. La bascule *B* doit attendre que la sortie *A* passe de 1 à 0 pour être déclenchée; la bascule *C* doit attendre la réaction de la sortie *B*, et ainsi de suite. Il s'établit donc un retard entre la réponse de chaque bascule. Dans les bascules modernes, ce retard est généralement très faible (de l'ordre de 10 à 40 ns); quoiqu'il en soit, nous verrons qu'il y a des cas où cela est ennuyeux. À cause de son type de fonctionnement, on désigne souvent ce circuit sous le nom de **compteur à propagation**. Dans la suite de ce texte, nous emploierons indistinctement les expressions « compteur asynchrone » et « compteur à propagation ».

EXEMPLE 7-1

Le compteur de la figure 7-1 est à l'état 0000 quand commence à arriver les impulsions d'horloge. Quelque temps après, on arrête les impulsions d'horloge et on lit l'état 0011 sur les différentes sorties du compteur. Dites combien d'impulsions ont été délivrées au compteur.

Solution La première réponse qui nous vient immédiatement est 3, puisque 0011 est l'équivalent binaire de 3. Toutefois, compte tenu de l'énoncé de cette question, il est impossible de dire si le compteur a été recyclé ou non. C'est donc dire qu'il peut y avoir eu 19 impulsions, les 16 premières ramenant le compteur à 0000 et les trois suivantes l'amenant dans l'état 0011. Comme il peut y en avoir eu 35 (deux cycles plus 3 impulsions), ou 51, ou...

MODULO Le compteur de la figure 7-1 possède 16 états distincts (0000 à 1111), on dit alors que c'est un *compteur à propagation MODULO-16*. Rappelons que le modulo est toujours égal au nombre d'états occupés par le compteur pendant un cycle complet avant son recyclage à l'état initial. Le modulo est porté à une valeur plus élevée simplement en ajoutant des bascules au compteur. Donc:

$$\text{MODULO} = 2^N \tag{7-1}$$

où *N* est le nombre de bascules raccordées comme dans le montage de la figure 7-1.

EXEMPLE 7-2

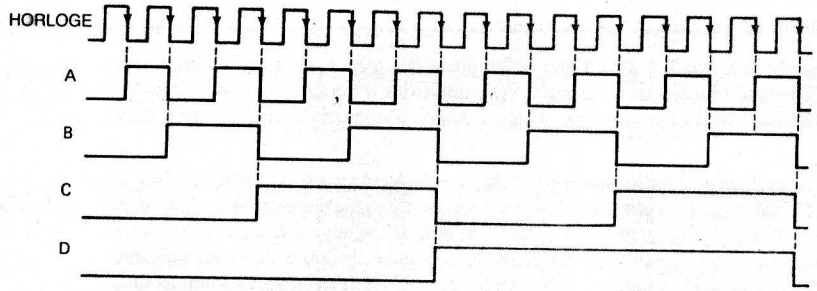
Il nous faut un compteur qui puisse dénombrer les articles circulant sur une bande de convoyeur. On a recours à une cellule photoélectrique et à une source lumineuse pour produire une impulsion chaque fois qu'un article coupe le faisceau. Ce compteur doit pouvoir compter jusqu'à mille articles. Dites combien il faut de bascules.

Solution Il est très simple de calculer la valeur *N* telle que $2^N \geq 1000$. Comme $2^9 = 512$, il nous faut manifestement plus de 9 bascules; $2^{10} = 1024$, c'est la solution cherchée. Il faut dix bascules pour compter jusqu'à $1023_{10} = 1111111111_2$. Nous pourrions

utiliser plus de dix bascules, mais ce serait du gaspillage car aucune des bascules au-delà de la dixième ne sera déclenchée.

Division de la fréquence Au chapitre 5, nous avons vu que chaque bascule d'un compteur élémentaire donnait en sortie une forme d'onde dont la fréquence est la moitié de la fréquence du signal d'horloge. À titre d'illustration, supposons que la fréquence du signal d'horloge de la figure 7-1 est 16 kHz. Les formes d'ondes correspondantes sont reproduites à la figure 7-2. La forme d'onde à la sortie *A* est une *onde carrée* de 8 kHz, celle à la sortie *B* une onde de 4 kHz, celle à la sortie de *C* une onde de 2 kHz et celle à la sortie *D* une onde de 1 kHz. Notez que le signal prélevé à la sortie *D* a une fréquence égale à la fréquence initiale divisée par 16. En règle générale, *la sortie de la dernière bascule d'un compteur est une onde dont la fréquence est celle du signal d'horloge d'entrée divisée par le MODULO du compteur*. Par exemple, on pourra appeler un compteur MODULO-16 un *compteur diviseur par 16*.

FIGURE 7-2 Formes d'ondes illustrant la division par deux de la fréquence occasionnée par chaque bascule.



EXEMPLE 7-3

La première étape de la construction d'une horloge numérique est le prélèvement d'un signal de 60 Hz du secteur et l'introduction de ce dernier dans un circuit conformateur d'impulsions pour obtenir une onde carrée comme celle de la figure 7-3. Cette onde carrée de 60 Hz est ensuite délivrée à un compteur MODULO-60, qui divise la fréquence initiale exactement par 60 pour produire un signal de 1 Hz. Ce dernier est alors appliqué aux bornes d'entrée d'une série de compteurs qui relèvent les secondes, les minutes, les heures. Dites combien il faut de bascules pour construire le compteur MODULO-60.

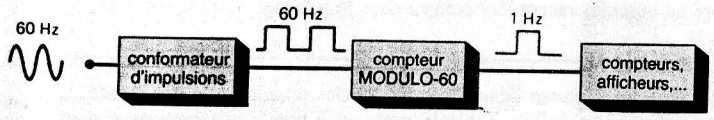


FIGURE 7-3 Exemple 7-3.

Solution Il n'existe pas de nombre entier tel que 2 élevé à sa puissance donne 60. Le résultat le plus près est $2^6 = 64$. Mais un compteur comportant six bascules divise par 64, ce qui, de toute évidence, n'est pas ce qu'on cherche. Il semble bien qu'il n'y ait pas moyen de résoudre notre problème en utilisant un compteur comme celui de la figure 7-1. Ce n'est que partiellement vrai; nous verrons dans la section suivante comment il est possible de modifier le compteur binaire de base pour obtenir le MODULO que l'on veut et, ainsi, ne plus être limité aux seules valeurs 2^N.

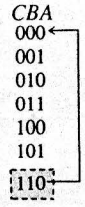
QUESTIONS DE RÉVISION

1. *Vrai ou faux:* Toutes les bascules d'un compteur asynchrone changent en même temps.
2. Supposez que le compteur de la figure 7-1 mémorise le nombre 0101. Dites quel sera le nombre stocké après 27 impulsions.

7-2 COMPTEURS À MODULO $< 2^N$

Les compteurs à propagation élémentaires comme ceux de la figure 7-1 ne peuvent avoir de MODULO différents de 2^N, où N est le nombre de bascules. Cette valeur est en réalité le MODULO maximal que l'on peut réaliser avec N bascules. Il est possible de modifier ce compteur élémentaire pour obtenir des MODULO inférieurs à 2^N en permettant au compteur de *sauter* certaines sections de la suite des nombres binaires. L'une des méthodes les plus courantes pour parvenir à ceci est illustrée à la figure 7-4 montrant un compteur à propagation de 3 bits. Pour le moment, ne tenons pas compte de la porte NON-ET; on peut donc voir que c'est un compteur binaire MODULO-8 qui affiche la suite des nombres de 000 à 111. Toutefois, comme nous allons le voir, la porte NON-ET perturbe cette suite de la manière suivante:

1. La sortie de la porte NON-ET est connectée aux entrées *RAZ* de chaque bascule. Tant que cette sortie reste à 1, le compteur n'est pas affecté. Lorsqu'elle devient 0, toutes les bascules sont ramenées à 0 et le compteur se remet immédiatement à compter à partir de l'état 000.
2. Les entrées de la porte NON-ET sont les sorties des bascules *B* et *C* de sorte que la sortie de cette porte passe à 0 lorsque $B = C = 1$. Cette condition survient quand le compteur passe de l'état 101 à l'état 110 (à la sixième impulsion). Le niveau BAS de la porte NON-ET place immédiatement (généralement en quelques nanosecondes) le compteur à l'état 000. Dès que les bascules ont été mises à 0, la sortie de la porte revient à 1, puisque la condition $B = C = 1$ n'existe plus.
3. La suite des nombres est donc



(état temporaire nécessaire à la remise à zéro du compteur)

Bien que le compteur passe par l'état 110, c'est à peine pour quelques nanosecondes avant son recyclage à 000. On peut donc dire que ce compteur va de 000 (zéro) à 101 (cinq) puis est recyclé à 000. En réalité, il saute les états 110 jusqu'à 111 de sorte qu'il n'affiche vraiment que six états différents; c'est donc un compteur MODULO-6.

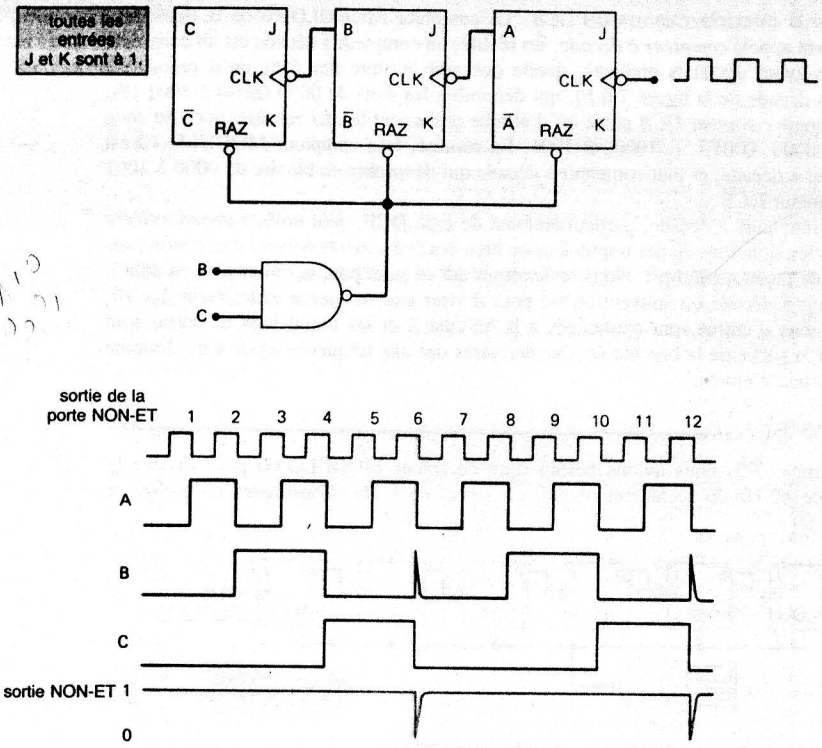


FIGURE 7-4 Compteur MODULO-6 obtenu en ramenant à 0 un compteur MODULO-8 quand le contenu du registre atteint six (110).

Remarquez que la forme d'onde de la sortie *B* est formée d'un *décrochement* ou d'un *signal transitoire* provoqué par l'apparition momentanée de l'état 110 avant la remise à zéro du compteur. Ce transitoire est très étroit et n'a aucun effet visible sur les lampes témoins ou les afficheurs numériques. Toutefois, il peut provoquer des difficultés si la sortie de *B* sert à piloter d'autres circuits du compteur. Notez que la fréquence de *C* égale un sixième de la fréquence d'entrée; autrement dit, le compteur MODULO-6 divise la fréquence du signal d'entrée par *six*. La forme d'onde du signal de *C* n'est *pas* une onde carrée symétrique (ayant un facteur d'utilisation de 50 pour cent) du fait que le niveau HAUT n'est présent que pendant deux impulsions tandis que le niveau BAS est présent pendant quatre impulsions.

EXEMPLE 7-4

Dessinez le graphe des états du compteur de la figure 7-4.

Solution La figure 7-5 nous montre comment les bascules *C*, *B* et *A* changent d'états à mesure qu'arrivent les impulsions sur l'entrée *CLK* de la bascule *A*. Rappelons que chaque cercle du graphe correspond à un des états possibles du compteur, et que les flèches figurent une transition d'un état à l'autre à l'instant d'une impulsion d'horloge.

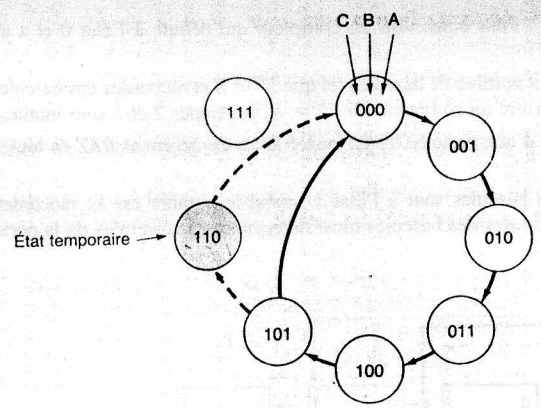


FIGURE 7-5 Graphe des états du compteur MODULO-6 de la figure 7-4.

Si nous supposons que le nombre initial est 000, le graphe illustre le passage normal d'un état à l'autre jusqu'au nombre 101. À l'instant du signal d'horloge suivant, le compteur passe temporairement dans l'état 110 avant de se mettre dans l'état stable 000. Les lignes en tiretées indiquent la nature temporaire de l'état 110. Comme nous l'avons déjà dit, la durée de l'état temporaire est tellement courte que, pratiquement, on peut considérer que le compteur passe directement de l'état 101 à 000 (flèche pleine).

Notez que l'état 111 n'est jamais réalisé, même temporairement.

Changement du MODULO Le modulo du compteur de la figure 7-4 est 6, et cela est conditionné par les entrées qu'on a choisies pour la porte NON-ET. Il est possible de changer ce MODULO en changeant les entrées. Par exemple, en utilisant une porte NON-ET à trois entrées, et en appliquant à celles-ci les sorties *A*, *B* et *C*, le compteur dénombre les états jusqu'à la condition 111, rendu à ce point il revient immédiatement à l'état 000. Donc si on ne tient pas compte de l'état momentané 111, le compteur parcourt les états 000 à 110 avant d'être recyclé à 000; c'est donc un compteur MODULO-7 (sept états).

EXEMPLE 7-5

Déterminez le MODULO du compteur de la figure 7-6(a). En outre, calculez la fréquence de la sortie *D*.

Solution Nous avons là un compteur de 4 bits, qui normalement parcourt la suite des nombres de 0000 à 1111. Les entrées de la porte NON-ET sont *D*, *C* et *B*, ce qui signifie que le compteur est recyclé immédiatement à 0 lorsqu'il atteint l'état 1110 (14). Donc ce compteur a 14 états stables de 0000 jusqu'à 1101, et c'est donc un compteur **MODULO-14**. Étant donné que la fréquence d'entrée est 30 kHz, la fréquence de sortie de *D* sera:

$$\frac{30\text{ kHz}}{14} = 2{,}14\text{ kHz}$$

Marche à suivre générale Pour construire un compteur qui débute à l'état 0 et a un MODULO X:

1. Trouvez le plus petit nombre de bascules tel que $2^N \geq X$ et raccordez ces bascules de manière à construire un compteur. Si $2^N = X$, les étapes 2 et 3 sont inutiles.
2. Connectez la sortie d'une porte NON-ET aux entrées asynchrones *RAZ* de toutes les bascules.
3. Déterminez quelles bascules sont à l'état 1 quand le nombre est X; raccordez alors les sorties normales des bascules ainsi déterminées aux entrées de la porte NON-ET.

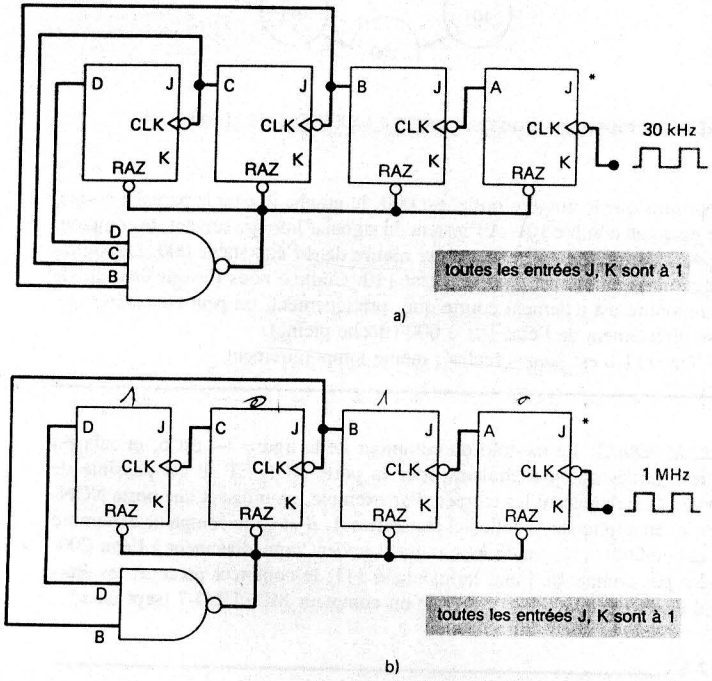


FIGURE 7-6 a) Compteur à propagation MODULO-14; b) Compteur à propagation MODULO-10 (à décade).

EXEMPLE 7-6

Construisez un compteur MODULO-10 qui dénombre les états de 0000 (zéro) à 1001 (9).

Solution $2^3 = 8$ et $2^4 = 16$; donc il nous faut quatre bascules. Comme le compteur doit avoir un fonctionnement stable jusqu'au nombre 1001, il doit être ramené à 0 lorsqu'il atteint le nombre 1010. C'est donc dire qu'il faut raccorder les sorties D et B des bascules aux entrées de la porte NON-ET. On peut voir ce montage à la figure 7-6 b).

Compteurs à décade/Compteurs DCB Le compteur MODULO-10 de la figure 7-6 est également appelé *compteur à décade*. En réalité, un compteur à décade est un compteur quelconque ayant dix états distincts, quelle que soit la suite des états qu'il prend. Le compteur à décade de la figure 7-6 b), qui dénombre les états de 0000 (zéro) à 1001 (9), est aussi appelé *compteur DCB* parce qu'il affiche seulement les dix représentions du code DCB, soit 0000, 0001, ..., 1000, et 1001. En résumé, tout compteur MODULO-10 est un compteur à décade, et tout compteur à décade qui dénombre en binaire de 0000 à 1001 est un compteur DCB.

Les compteurs à décade, particulièrement de type DCB, sont utilisés abondamment dans toutes les situations où des impulsions ou bien des événements doivent être dénombrés et affichés de façon numérique. Nous reviendrons sur ce sujet pour le traiter plus en détail. Un compteur à décade est souvent utilisé pour diviser une fréquence *exactement* par 10. Les impulsions d'entrée sont appliquées à la bascule A et les impulsions de sortie sont prélevées à la sortie de la bascule D. Ces dernières ont une fréquence égale à un dixième de la fréquence d'entrée.

EXEMPLE 7-7

Dans l'exemple 7-3, nous avions besoin d'un compteur MODULO-60 pour diviser la fréquence de 60 Hz du secteur et obtenir un signal de 1 Hz. Construisez ce compteur MODULO-60.

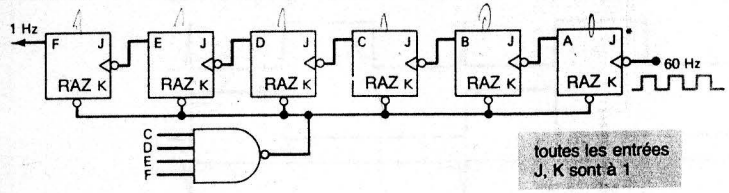


FIGURE 7-7 Compteur MODULO-60.

Solution $2^5 = 32$ et $2^6 = 64$; donc nous avons besoin de six bascules, raccordées comme à la figure 7-7. Le compteur est réinitialisé lorsqu'il atteint le nombre 60 (111100). Donc les sorties des bascules qu'il faut raccorder aux entrées de la porte NON-ET sont C, D, E et F. La fréquence de la sortie F sera 1 Hz.

QUESTIONS DE RÉVISION

1. Dites quelles sorties des bascules doivent-être raccordées à la porte NON-ET de remise à zéro pour qu'on obtienne un compteur MODULO-13.
2. *Vrai ou faux:* Tous les compteurs DCB sont des compteurs à décade.
3. Dites quelle est la fréquence de sortie d'un compteur à décade synchronisé par un signal de 50 kHz.

7-3 COMPTEURS ASYNCHRONES CI

Il existe de nombreux compteurs intégrés asynchrones, autant en technologie TTL qu'en technologie CMOS. L'un d'eux est le boîtier TTL 7493 (74LS93...), de même que son équivalent logique le 74293 (74LS293...) Le 74293 est préféré dans les nouveaux pro-

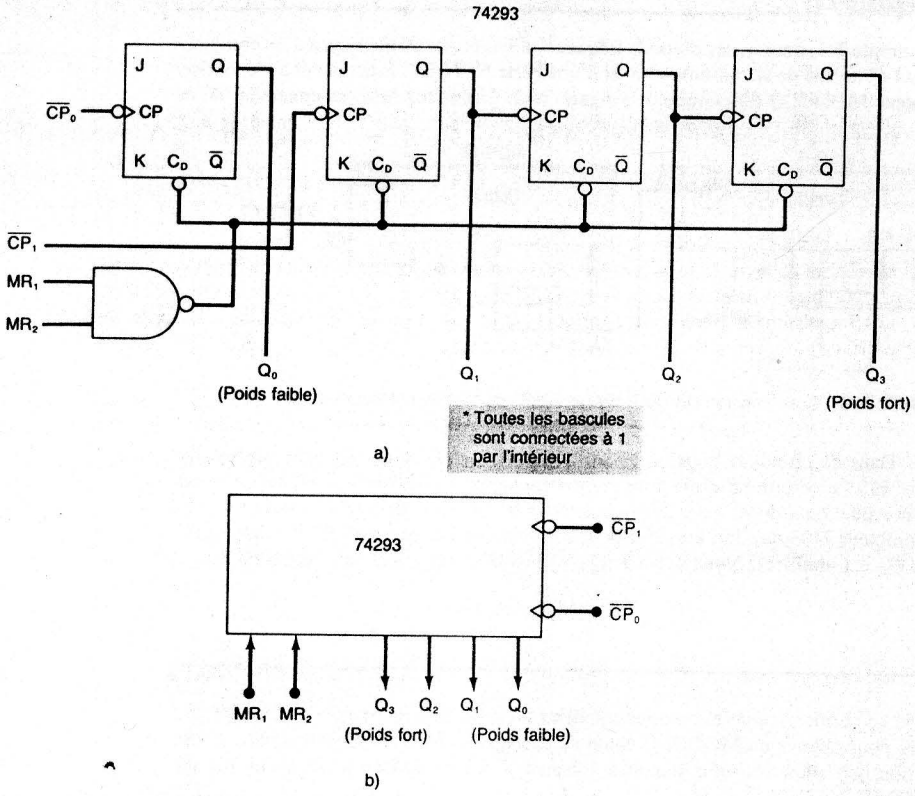


FIGURE 7-8 a) Le schéma logique du compteur asynchrone intégré 7493 ou 74293; b) symbole simplifié.

jets parce que ses broches d'alimentation et de masse sont aux angles du boîtier. La figure 7-8 a) montre le schéma logique du 74293, tel qu'on peut le voir dans la fiche technique TTL du fabricant. Certains termes sur ce schéma sont différents de ceux qu'on a utilisés jusqu'ici, mais la correspondance est facile à établir. Nous pouvons relever les points suivants:

1. Le 74293 contient quatre bascules J-K dont les sorties sont Q_0, Q_1, Q_2 et Q_3 où Q_0 est le rang de poids faible et Q_3 est le rang de poids fort. Les bascules sont couplées pour que le rang de poids faible soit à gauche. On procède ainsi pour respecter la convention qui veut que le signal d'entrée arrive à gauche. Nous avons dessiné jusqu'à présent nos bascules avec le rang de poids faible à droite de façon à avoir l'ordre des bascules dans le même ordre que les bits d'un nombre binaire. Nous continuerons de les placer ainsi.
2. Chaque bascule dispose d'une entrée *CP* (signal d'horloge), autre désignation pour l'entrée *CLK*. Les entrées d'horloge de Q_0 et Q_1, *libellées respectivement* $\overline{CP}_0$ et $\overline{CP}_1$, sont accessibles par l'extérieur. Les barres d'inversion sur ces entrées signalent des entrées actives sur leur front descendant.

3. Chaque bascule possède une entrée asynchrone de REMISE À ZÉRO, C_D. Ces entrées sont toutes raccordées à la sortie d'un porte NON-ET à deux entrées, MR_1 et MR_2; ces deux entrées sont des bornes de *remise à zéro principale* (*Master Reset*).
4. Les bascules Q_1, Q_2 et Q_3 sont déjà connectées comme un compteur à propagation de 3 bits. La bascule Q_0 n'est raccordée à *rien* par l'intérieur. Cela permet à l'utilisateur de relier Q_0 à Q_1 pour former un compteur de 4 bits, ou de laisser Q_0 isolée.

Les exemples suivants nous font voir différentes façons de câbler la puce 74293 pour réaliser des compteurs. Dans ces exemples, nous utilisons le symbole logique simplifié introduit à la figure 7-8 b).

EXEMPLE 7-8

Montrez comment raccorder les bornes de la puce 74293 pour obtenir un compteur MODULO-16 synchronisé par un signal de 10 kHz.

Solution Pour construire un compteur MODULO-16, il nous faut quatre bascules; nous devons donc raccorder la sortie Q_0 à $\overline{CP}_1$, l'entrée du signal d'horloge de la bascule Q_1 (voir figure 7-9). Les impulsions de 10 kHz sont appliquées à $\overline{CP}_0$, l'entrée du signal d'horloge de Q_0.

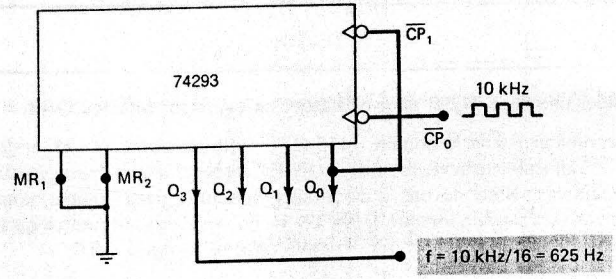


FIGURE 7-9 Une puce 74293 câblée pour constituer un compteur MODULO-16.

Pour que ce compteur fonctionne avec un MODULO-16, il doit parcourir toute la suite 0000 à 1111. Il faut donc que la porte NON-ET de réinitialisation soit invalidée; c'est-à-dire que sa sortie soit toujours à 1, pour qu'elle ne vienne pas affecter le fonctionnement du compteur. C'est ce qu'on fait en raccordant MR_1 et MR_2 en permanence à 0.

EXEMPLE 7-9

Montrez comment câbler la puce 74293 pour constituer un compteur MODULO-10.

Solution Pour réaliser un compteur MODULO-10, il nous faut quatre bascules, et nous devons, de nouveau, connecter Q_0 à $\overline{CP}_1$. Cependant cette fois il faut que le compteur soit recyclé à 0000 lorsqu'il tente de passer au nombre 1010 (10). Donc les sorties Q_3 et Q_1 doivent être connectées aux entrées initialisation principale; lorsque l'arrivée du nombre 1010 met à 1 ces deux entrées, la sortie de la porte NON-ET remet immédiatement le compteur à 0000.

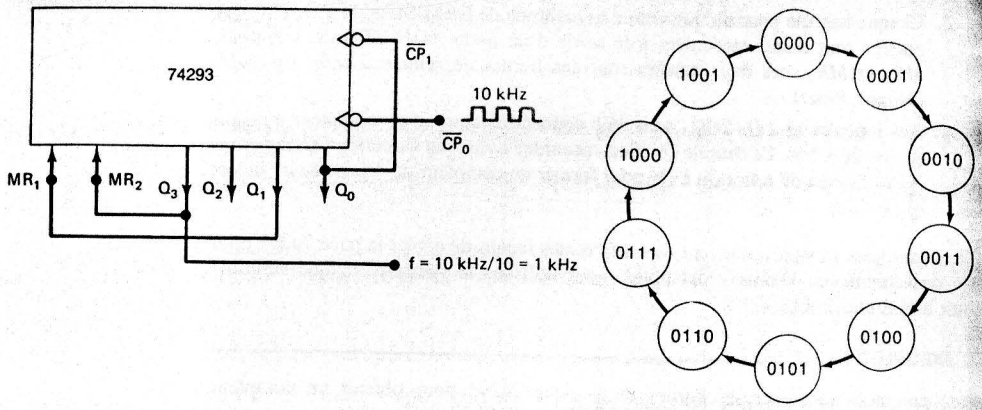


FIGURE 7-10 Une puce 74293 câblée pour constituer un compteur MODULO-10.

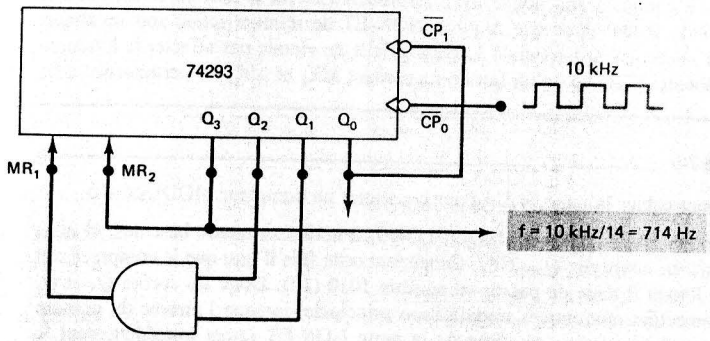
On montre à la figure 7-10 le câblage de ce circuit. On peut y voir aussi le graphe de transition. Remarquez l'absence de l'état temporaire 1010.

EXEMPLE 7-10

Montrez comment câbler une puce 74293 pour constituer un compteur MODULO-14.

Solution Quand le compteur atteint le nombre 1110 (14), les trois sorties Q_3, Q_2 et Q_1 sont au niveau HAUT. Malheureusement, la porte NON-ET intégrée dans la puce 74293 ne contient que deux entrées. Nous devons donc ajouter une autre porte logique pour s'assurer que le compteur sera recyclé lorsque $Q_3 = Q_2 = Q_1 = 1$. En fait tout ce qu'il nous faut c'est une autre porte ET à deux entrées comme celle de la figure 7-11.

FIGURE 7-11 Pour réaliser un compteur MODULO-14 avec une puce 74293, il faut ajouter une porte ET.



EXEMPLE 7-11

Dans l'exemple 7-7, nous avons divisé la fréquence d'entrée par 60 au moyen d'un compteur MODULO-60 formé de six bascules *J-K* et d'une porte NON-ET. Il est possible de réaliser un compteur MODULO-60 comme à la figure 7-12. Expliquez le fonctionnement de ce circuit.

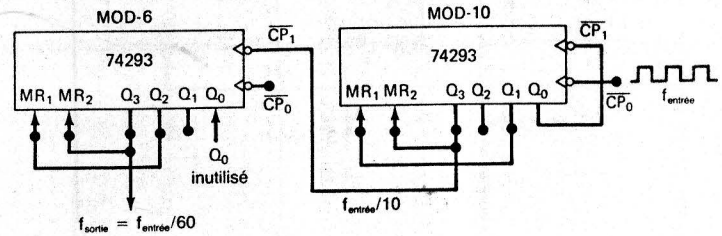


FIGURE 7-12 Combinaison de deux puces 74293 qui divise la fréquence par 60.

Solution Dans ce circuit la division de la fréquence par 60 se fait en deux étapes. Le compteur 74293 de droite est câblé pour réaliser un compteur MODULO-10, de sorte que la fréquence du signal de sortie Q_3 est $f_{entrée}/10$. Ce signal est délivré à l'entrée $\overline{CP}_1$ du second compteur 7493 qui, lui, est câblé pour constituer un compteur MODULO-6 (remarquez que Q_0 est inutilisé). Donc la sortie Q_3 du second compteur a une fréquence de

$$f_{\text{sortie}} = \frac{f_{\text{entrée}}/10}{6} = \frac{f_{\text{entrée}}}{60}$$

Dans ce dernier exemple, on a vu comment il est possible de mettre en cascade des compteurs pour obtenir un MODULO égale au *produit* des MODULO individuels. C'est une solution que vous trouverez très utile lorsqu'il s'agit de diviser la fréquence par un grand nombre.

Symbole IEEE/ANSI pour le compteur 74293 La figure 7-13 reproduit le symbole IEEE/ANSI du 74293. Ce symbole illustre plusieurs aspects inédits de la norme IEEE/ANSI. Pendant notre description, vous constaterez comment la nouvelle symbologie IEEE/ANSI a été élaborée pour révéler le plus de choses possibles sur le fonctionnement de ce circuit intégré.

On discerne dans ce symbole trois blocs distincts. Le bloc supérieur (avec les renfoncements) s'appelle le bloc de contrôle commun. Le libellé « CTR » désigne ce circuit intégré comme étant un compteur. Comme nous l'avons signalé au chapitre 5, le bloc de contrôle commun sert dans un CI chaque fois qu'une ou plusieurs entrées sont communes à tous les circuits se trouvant dans la puce. Pour le 74293, les entrées MR_1 et MR_2 sont communes à toutes les bascules du compteur.

Les bornes MR_1 et MR_2 sont illustrées comme des entrées vraies au niveau HAUT combinées à l'intérieur dans une porte ET (ce que nous apprend le symbole « & »). Cela nous permet de déduire que les deux entrées MR_1 et MR_2 doivent être à leur valeur vraie pour mettre à zéro le compteur. Le libellé « CT = 0 » nous informe que l'action des entrées *MR* est de ramener le nombre dans le compteur à zéro.

Le bloc du milieu est appelé « DIV2 » pour signifier qu'il s'agit d'un compteur MODULO 2, en réalité une simple bascule. DIV2 signifie que le compteur divise la fréquence du signal d'entrée par 2. Le bloc du bas est libellé « DIV8 » ce qui correspond

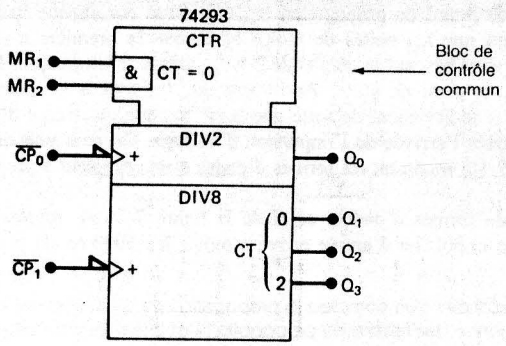


FIGURE 7-13 Symbole IEEE/ANSI du CI 74293.

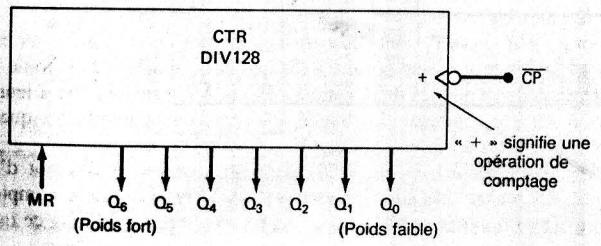
à un compteur MODULO 8. Les signaux d'horloge de ces blocs nous apprennent qu'ils sont actifs pendant leurs fronts descendants. Le signe « + » à la suite de chaque entrée d'horloge veut nous informer qu'à chaque front descendant de l'horloge le compteur est *incrémenté de 1*. Autrement dit, le compteur *progresse* à chaque front descendant. Un signe « − » aurait désigné un *décompteur* (voyez la section 7-4).

Bien que nous allons continuer de nous servir du symbole standard pour le 74293 et les compteurs standard, nous n'hésiterons pas à nous approprier certains éléments de la symbologie IEEE/ANSI. Par exemple, nous indiquerons le MODULO d'un compteur en empruntant la notation DIV*n*, où *n* est le MODULO.

Compteurs asynchrones CMOS La famille CMOS regroupe de nombreux compteurs asynchrones. La plupart d'entre eux possèdent une version équivalente dans la technologie TTL. Il se trouve, toutefois, des compteurs intégrés CMOS qui n'ont pas leur contrepartie TTL. L'un de ceux-ci est la puce 74HC4024 dont le symbole logique est reproduit à la figure 7-14. Il s'agit d'un compteur de 7 bits possédant une entrée asynchrone de remise à zéro principale. Les sept bascules sont agencées en interne pour constituer un compteur à propagation MODULO 128. L'entrée *MR* est vraie au niveau HAUT; elle sert à remettre à zéro toutes les bascules. Remarquez le libellé « CTR DIV128 » qui désigne un compteur MODULO-128.

Un autre compteur à propagation qui n'a pas d'équivalent en technologie TTL est le 74HC4040, un compteur 12 bits possédant une seule entrée principale de remise à zéro. Le signal d'horloge de ce compteur est une entrée de type Schmitt; donc avec ce compteur, même des signaux d'horloge à variation lente ne donnent pas lieu à un comptage erratique.

FIGURE 7-14 Symbole logique du compteur à propagation CMOS 74HC4024.



QUESTIONS DE RÉVISION

1. Un signal d'horloge de 2 kHz est appliqué à l'entrée $\overline{CP}_1$ d'un 74293. Quelle est la fréquence en Q_3?
2. Quelle devrait être la fréquence finale si l'ordre des compteurs de la figure 7-12 est inversé?
3. Quel est le MODULO d'un compteur 74HC4040?
4. Que signifie le libellé « DIV64 » dans un symbole de compteur?

7-4 DÉCOMPTEURS ASYNCHRONES

Tous les compteurs étudiés jusqu'à présent comptaient *progressivement* à partir de zéro; c'étaient donc des *compteurs progressifs*. Il est relativement facile de construire des *décompteurs* asynchrones (à propagation) dont les états régressent, c'est-à-dire qui partent d'un nombre maximal pour arriver jusqu'à zéro. Avant d'aborder l'étude d'un décompteur à propagation, examinons la suite des nombres inverses pour un décompteur de 3 bits:

	CBA	*CBA*	*CBA*
(7)	1 1 1	1 1 1	1 1 1
(6)	1 1 0	1 1 0	1 1 0
(5)	1 0 1	1 0 1	. . .
(4)	1 0 0	1 0 0	. . .
(3)	0 1 1	0 1 1	. . .
(2)	0 1 0	0 1 0	etc.
(1)	0 0 1	0 0 1	
(0)	0 0 0	0 0 0	etc.

A, *B* et *C* représentent les états de sortie des bascules à mesure que le compteur descend. On peut voir que la bascule *A* (poids faible) change d'état (commute) à chaque ligne de cette suite, comme dans le cas du compteur progressif. La bascule *B* commute chaque fois que *A* passe de 0 à 1; *C* commute chaque fois que *B* passe de 0 à 1. Ainsi, dans un décompteur, chaque bascule, sauf la première, passe à l'état opposé quand la bascule qui la précède effectue la transition de 0 à 1. Si les bascules du compteur changent d'états lorsque passe un front descendant du signal d'horloge, il faut placer un inverseur avant chaque entrée *CLK*. Il est toutefois possible d'arriver au même résultat en pilotant chaque entrée *CLK* des bascules au moyen de la sortie *complémentée* de la bascule qui précède. C'est ce que nous montre la figure 7-15 dans le cas d'un décompteur MODULO-8.

Les impulsions d'entrée sont appliquées à la bascule *A*; la sortie $\overline{A}$ est ramenée à l'entrée *CLK* de la bascule *B*; la sortie $\overline{B}$ est ramenée à l'entrée *CLK* de la bascule *C*. L'examen des formes d'ondes de *A*, *B* et *C* montre que *B* commute lorsque *A* passe de 0 à 1 (donc $\overline{A}$ passe de 1 à 0) et que *C* commute quand *B* passe de 0 à 1. De cette manière nous obtenons une suite de nombres régressive sur les sorties *C*, *B* et *A*. Le graphe des états illustre cette séquence. Comparez-le avec le graphe du compteur MODULO 8 de la figure 5-49.

Les décompteurs ne sont pas aussi répandus que les compteurs. On les utilise principalement pour indiquer l'arrivée d'un nombre fixé d'impulsions d'entrée. Dans de telles situations, le décompteur est *préréglé* avec le nombre recherché et la valeur qu'il mémorise décroît à mesure qu'arrivent les impulsions. Lorsque le compteur arrive à l'état *zéro*, une porte logique décèle cette condition ce qui indique que le nombre d'impulsions lancées est égal au nombre inscrit dans le compteur. Nous étudierons dans une section ultérieure les compteurs préréglables.

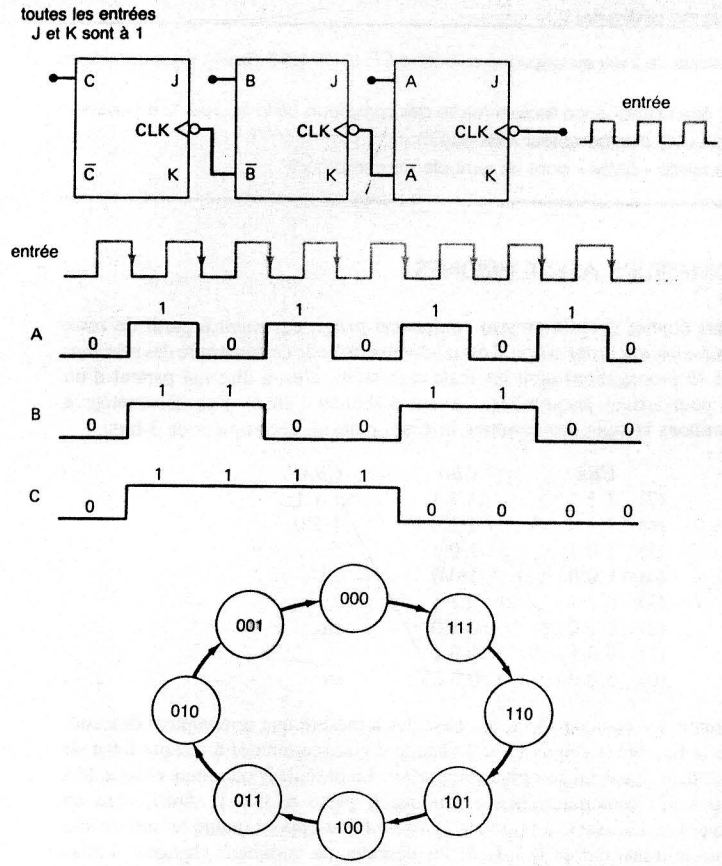


FIGURE 7-15 Décompteur MODULO-8.

QUESTIONS DE RÉVISION

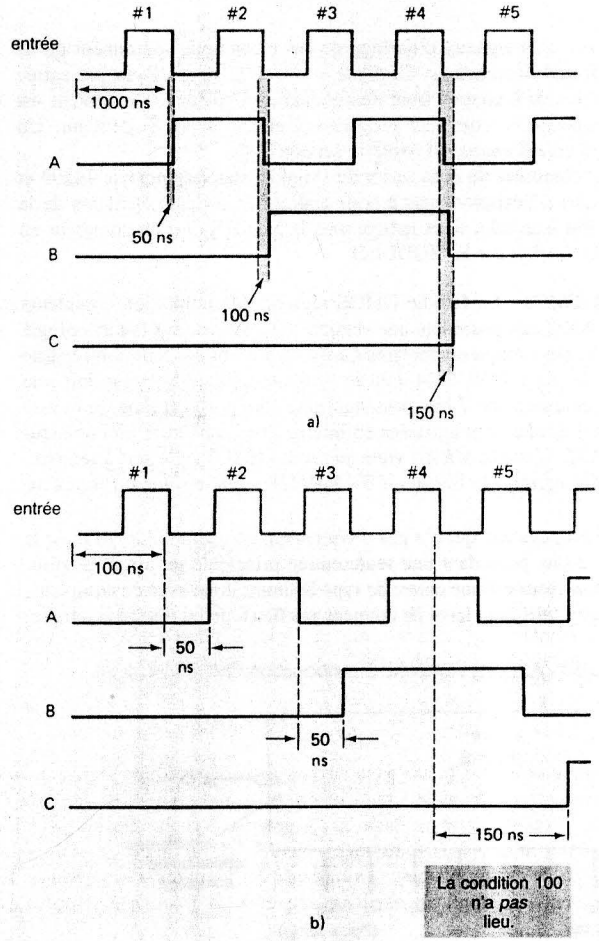
1. Dites quelle est la différence entre la suite des nombres d'un compteur progressif et celle d'un compteur régressif.
2. Indiquez la différence entre le circuit d'un décompteur asynchrone et le circuit d'un compteur.

7-5 RETARD DE PROPAGATION DANS LES COMPTEURS ASYNCHRONES

Les compteurs à propagation sont les types de compteurs binaires les plus simples, puisque ce sont eux qui exigent le moins de composants pour réaliser une certaine opération de comptage. Ils ont cependant un inconvénient majeur dû au principe fondamental de leur fonctionnement. Chaque bascule est déclenchée par la transition de la sortie de la bascule qui la précède. À cause du retard de propagation (t_{pd}) introduit par chaque bascule, la deuxième bascule ne réagit que t_{pd} unités de temps après que la première a reçu une impulsion d'entrée; la troisième bascule ne réagit que $2 \times t_{pd}$ unités de temps après l'arrivée de l'impulsion d'horloge, et ainsi de suite. Autrement dit, les retards de propagation introduits par les bascules s'additionnent de sorte que la N^e bascule ne change d'état que $N \times t_{pd}$ unités de temps après l'arrivée de l'impulsion d'horloge. On peut voir une illustration de ceci à la figure 7-16, montrant les formes d'ondes d'un compteur à propagation de 3 bits.

Le premier groupe de formes d'ondes, celui de la figure 7-16 a), représente une situation dans laquelle une impulsion d'entrée arrive à toutes les 1000 ns (la période de

FIGURE 7-16 Formes d'ondes d'un compteur à propagation de 3 bits illustrant la conséquence des retards de propagation des bascules pour deux fréquences d'entrée différentes.



l'horloge T = 1000 ns); on présume que le retard de propagation de chaque bascule est 50 ns (t_{pd} = 50 ns). Remarquez que la sortie de la bascule A commute 50 ns après le front descendant de l'impulsion d'entrée. De même, B commute 50 ns après que la sortie A a passé de 1 à 0, et C commute 50 ns après que la sortie B a passé de 1 à 0. Au total, quand arrive la quatrième impulsion d'entrée, la sortie C passe au niveau HAUT après un retard de 150 ns. Dans cette situation, le compteur fonctionne convenablement puisque les bascules passent éventuellement aux bons états qui représentent le nombre binaire. Toutefois, la situation empire si les impulsions d'entrée sont appliquées à un rythme beaucoup plus élevé.

Les formes d'ondes de la figure 7-16 b) nous font voir ce qui arrive si les impulsions d'entrée sont délivrées au rythme d'une à toutes les 100 ns. De nouveau, chaque sortie de bascule réagit 50 ns après la transition 1 − 0 du signal appliqué à l'entrée *CLK* (notez le changement du rapport relatif de ces temps). Il est particulièrement intéressant d'étudier la situation lorsque survient le front descendant de la quatrième impulsion d'entrée; la sortie C ne passera pas au niveau HAUT avant 150 ns, exactement au même moment où la sortie A passera au niveau HAUT en réponse à la cinquième impulsion d'entrée. En d'autres mots, la condition $C = 1$, $B = A = 0$ (nombre 100) ne survient jamais parce que la fréquence d'entrée est trop élevée. Ceci peut causer de graves problèmes si cette dernière condition doit commander d'autres opérations du système numérique. Il est possible d'éviter de faire face à de tels problèmes si on choisit des signaux d'entrée dont les périodes sont beaucoup plus grandes que le retard de propagation total du compteur. C'est-à-dire

$$T_{\text{horloge}} \geq N \times t_{pd} \tag{7-2}$$

où N = nombre de bascules. En termes de fréquence, on dit que la fréquence maximale utilisable est:

$$f_{max} = \frac{1}{N \times t_{pd}} \tag{7-3}$$

Par exemple, supposons qu'on construise un compteur à propagation de 4 bits avec des bascules J-K 74LS112. Le tableau 5-1, nous apprend que les retards de propagation entre *CLK* et Q pour la 74LS112 sont t_{PLH} = 16 ns et t_{PHL} = 24 ns. Pour calculer f_{max}, nous retiendrons « l'hypothèse la plus pessimiste »; c'est-à-dire que nous utiliserons $t_{pd} = t_{PHL}$ = 24 ns, ce qui donne

$$f_{max} = \frac{1}{4 \times 24 \text{ ns}} = 10{,}4 \text{ MHz}$$

Évidemment, à mesure qu'augmente le nombre de bits d'un compteur, le retard de propagation total augmente et f_{max} diminue. Par exemple, un compteur à propagation constitué de six bascules 74LS112 aura comme fréquence maximale

$$f_{max} = \frac{1}{6 \times 24 \text{ ns}} = 6{,}9 \text{ MHz}$$

On voit donc que les compteurs asynchrones ne sont pas d'une grande utilité aux très hautes fréquences, particulièrement lorsqu'ils ont un grand nombre de bits. Un autre problème lié au retard de propagation dans les compteurs asynchrones survient lorsque les sorties du compteur sont *décodées*. C'est un problème que nous aborderons dans une section ultérieure. En dépit de ces problèmes, la simplicité des compteurs asynchrones les destine tout particulièrement aux applications dans lesquelles la fréquence limite n'est pas un élément critique.

QUESTIONS DE RÉVISION

1. Expliquez comment il se fait que la fréquence limite d'un compteur à propagation change à mesure qu'on ajoute des bascules.
2. Une certaine bascule J-K a un t_{pd} = 12 ns. On veut construire un compteur avec des bascules de ce type qui fonctionnerait à 10 MHz, quel est le plus grand MODULO réalisable?

7-6 COMPTEURS SYNCHRONES (PARALLÈLES)

Les problèmes causés par les compteurs à propagation sont imputables aux retards de propagation de bascules montées en cascade; autrement dit, dans ces derniers compteurs les bascules ne changent pas d'état simultanément avec les impulsions d'entrée. On contourne cette limitation en utilisant des compteurs *parallèles* ou *synchrones* dans lesquels toutes les bascules sont déclenchées simultanément (en parallèle) par les impulsions d'horloge d'entrée. Étant donné que les impulsions d'horloge sont appliquées à toutes les bascules, il doit y avoir un certain mécanisme qui indique quand une impulsion d'horloge doit faire commuter une bascule ou la laisser dans le même état. On réalise un tel mécanisme en utilisant les entrées J et K; on peut voir un exemple de ceci à la figure 7-17 pour un compteur parallèle MODULO-16 à 4 bits.

Si on compare le montage de ce compteur parallèle avec celui de son équivalent asynchrone de la figure 7-1, on peut relever les différences suivantes:

- Les entrées *CLK* de toutes les bascules sont raccordées ensemble afin que le signal d'horloge arrive simultanément sur toutes les bascules.
- Seule la bascule A, le rang de poids faible, a ses entrées J et K en permanence au niveau HAUT. Les entrées J, K des autres bascules sont pilotées par une certaine combinaison des sorties des bascules.
- Le compteur parallèle nécessite plus d'éléments de circuit que le compteur asynchrone.

Fonctionnement du circuit Le principe de fonctionnement d'un compteur parallèle est celui-ci:

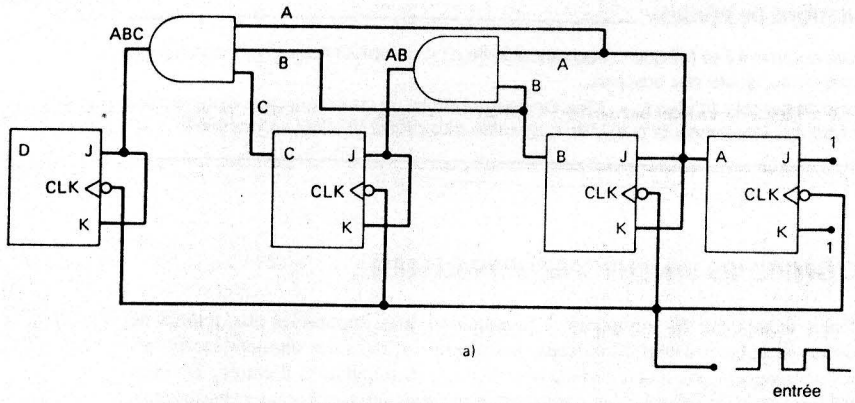
Les entrées J et K des bascules sont raccordées de façon à ce que les bascules qui doivent commutées sur un certain FD aient $J = K = 1$ quand ce FD survient.

Examinons la mise en pratique de ce principe pour chacune des bascules en nous aidant de la suite des nombres illustrée à la figure 7-17 b).

La suite des nombres comptés nous fait voir que la bascule A doit changer d'état à chaque front descendant. C'est pour cette raison que ses entrées J et K sont toujours gardées à 1 afin que cette bascule commute à chaque front descendant du signal d'horloge.

La suite des nombres comptés nous fait voir que la bascule B doit changer d'état à chacun des FD survenant quand $A = 1$. Par exemple, quand le nombre est 0001, le FD suivant fait basculer B vers l'état 1; quand le nombre est 0011, le FD suivant fait, cette fois-ci, basculer B vers l'état 0; et ainsi de suite. Ce fonctionnement est obtenu en raccordant la sortie de A aux entrées J et K de la bascule B, de sorte que $J = K = 1$ tant que $A = 1$.

La suite des nombres montre que la bascule C doit changer d'état à chacun des FD survenant quand $A = B = 1$. Par exemple, quand le nombre est 0011, le FD suivant doit



a)

Nombre	D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
0	0	0	0	0
.	.	.	.	.
.	.	.	.	.
.	.	etc.	.	.

b)

FIGURE 7-17 Chaque bascule est commandée par le FD du signal d'entrée, d'où le déclenchement simultané de toutes les bascules.

faire basculer C vers l'état 1; quand le nombre est 0111, le FD suivant doit faire basculer C, cette fois-ci, vers 0; et ainsi de suite. Ce fonctionnement est réalisé grâce au raccordement du signal AB sur les entrées J et K de la bascule C.

En suivant la même analyse, on constate que la bascule D doit commuter à chaque FD qui survient quand $A = B = C = 1$. Quand le nombre est 0111, le FD suivant doit faire basculer D vers l'état 1; quand le nombre est 1111, le FD suivant doit faire basculer D vers l'état 0. On s'assure que cela se produit en raccordant le signal ABC sur les entrées J et K de la bascule D.

Supériorité des compteurs parallèles sur les compteurs asynchrones Dans un compteur parallèle, toutes les bascules changent d'états en même temps; c'est-à-dire qu'elles sont synchronisées sur les FD des signaux de l'horloge. Ainsi, contrairement aux compteurs

asynchrones, les retards de propagation des bascules ne se combinent pas pour accroître le retard total. Au contraire, le temps de réponse total d'un compteur parallèle comme celui de la figure 7-17 est égal au retard d'une *seule* bascule plus le temps que les nouveaux niveaux logiques prennent pour franchir une *seule* porte ET et arriver sur les entrées J et K. Donc:

$$\text{retard total} = t_{pd} \text{ d'une bascule} + t_{pd} \text{ d'une porte ET}$$

Cette équation nous apprend qu'un compteur parallèle peut fonctionner à des fréquences de loin supérieures à celles d'un compteur asynchrone comptant le même nombre de bascules. Évidemment, le compteur synchrone demande plus d'éléments de circuit que le compteur asynchrone.

L'aptitude à fonctionner à des fréquences supérieures est un des grands atouts des compteurs parallèles. Un autre atout important concerne la solution de *décodage*, sujet que nous abordons dans une section ultérieure.

CI réels On retrouve plusieurs compteurs parallèles intégrés aussi bien dans la famille TTL que dans la famille CMOS. Certains des dispositifs les plus courants sont:

74LS160/162, 74HC160/162; compteurs à décade asynchrones
74LS161/163, 74HC161/163; compteurs MODULO-16 parallèles

EXEMPLE 7-12

a) Déterminez f_{max} pour le compteur de la figure 7-17 a) si t_{pd} de chaque bascule est 50 ns et t_{pd} de chaque porte ET est 20 ns. Comparez cette f_{max} à celle d'un compteur à propagation MODULO-16.
b) Dites ce qu'il faut faire pour convertir ce compteur en un compteur MODULO-32.
c) Déterminez f_{max} pour un compteur parallèle MODULO-32.

Solution

a) Le retard total minimal qu'il peut y avoir entre les impulsions d'entrée est égal à t_{pd} d'une bascule + t_{pd} d'une porte ET. Dont $T_{horloge} \geq 50 + 20 = 70$ ns, de sorte que pour le compteur parallèle

$$f_{max} = \frac{1}{70 \text{ ns}} = 14{,}3 \text{ MHz} \quad \text{(compteur parallèle)}$$

Un compteur à propagation MODULO-16 utilise quatre bascules ayant $t_{pd} = 50$ ns. Donc son f_{max} est:

$$f_{max} = \frac{1}{4 \times 50 \text{ ns}} = 5 \text{ MHz} \quad \text{(compteur à propagation)}$$

b) On doit ajouter une cinquième bascule car $2^5 = 32$. L'entrée CLK de cette bascule est également alimentée par les impulsions d'entrée. Ses entrées J et K sont alimentées par la sortie d'une porte ET à quatre entrées dont les entrées sont A, B, C et D.
c) f_{max} ne diffère pas de la valeur trouvée en a) malgré qu'il y ait plus de bascules dans ce compteur parallèle. Donc f_{max} est toujours 14,3 MHz.

QUESTIONS DE RÉVISION

1. Dites quel est l'avantage d'un compteur synchrone par rapport à un compteur asynchrone. Dites quel en est l'inconvénient.
2. Combien faut-il d'éléments logiques pour construire un compteur parallèle MODULO-64?

7-7 DÉCOMPTEURS ET COMPTEURS RÉVERSIBLES PARALLÈLES

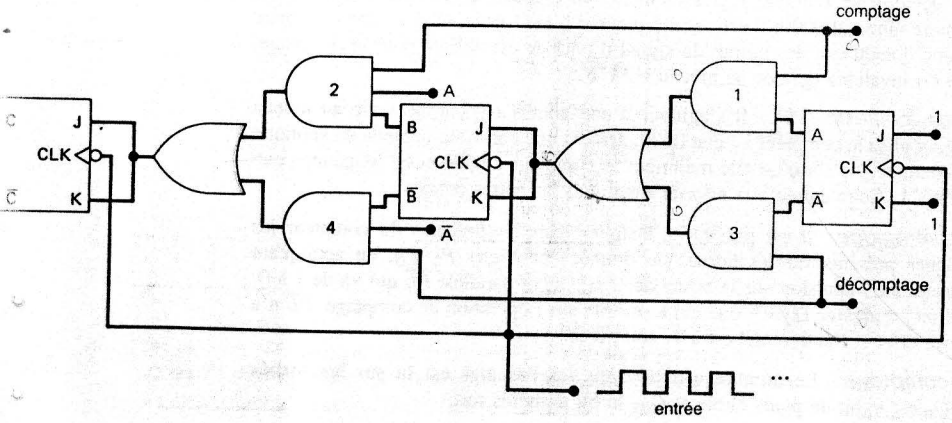
À la section 7-4, nous avons vu qu'on peut facilement transformer un compteur à propagation en un décompteur en utilisant les sorties complémentées de chaque bascule pour piloter la bascule suivante du compteur. De la même manière on peut construire un décompteur parallèle; il suffit d'utiliser les sorties complémentées des bascules pour alimenter les entrées *J-K* suivantes. Par exemple, le compteur parallèle de la figure 7-17 peut être transformé en un décompteur en connectant les sorties $\overline{A}$, $\overline{B}$ et $\overline{C}$ au lieu des sorties A, B et C sur la porte ET, respectivement. La suite des nombres affichés progressivement par ce compteur à mesure qu'arrivent les impulsions est:

(15) 1 1 1 1
(14) 1 1 1 0
(13) 1 1 0 1
(12) 1 1 0 0
. . . .
. . . . recyclage
. . . .
(3) 0 0 1 1
(2) 0 0 1 0
(1) 0 0 0 1
(0) 0 0 0 0

Pour réaliser un compteur/décompteur parallèle (voir figure 7-18) on utilise les entrées COMPTAGE ET DÉCOMPTAGE pour déterminer si on alimentera les entrées J et K de la bascule suivante avec les sorties normales ou complémentées de la bascule. Le compteur de la figure 7-18 est un compteur réversible MODULO-8 (dit également compteur-décompteur) qui dénombre les états de 000 jusqu'à 111 lorsque l'entrée de commande COMPTAGE est à 1 et qui dénombre régressivement les états 111 jusqu'à 000 lorsque l'entrée de commande DÉCOMPTAGE est à 1.

Une valeur logique 1 sur la ligne COMPTAGE et une valeur logique 0 sur la ligne DÉCOMPTAGE valident les portes ET 1 et 2 et invalident les portes ET 3 et 4. Ainsi les sorties A et B sont raccordées aux entrées J et K des bascules suivantes, de sorte que le

FIGURE 7-18 Compteur/décompteur parallèle MODULO-8.



compteur dénombre progressivement les impulsions introduites. L'action inverse se produit quand COMPTAGE = 0 et DÉCOMPTAGE = 1.

CI réels Les CI 74LS190 et 74HC190 sont des exemples des nombreux compteurs/décompteurs intégrés. Ces derniers sont des compteurs/décompteurs à décade.

EXEMPLE 7-13

Décrivez le fonctionnement du circuit de la figure 7-18 quand les entrées COMPTAGE et DÉCOMPTAGE sont toutes les deux à 0; quand elles sont toutes les deux à 1.

Solution Quand ces deux entrées sont à 0, toutes les portes ET et toutes les portes OU ont leurs sorties à 0. Ainsi les entrées *J-K* des bascules B et C sont à 0 et aucune de ces bascules ne peut changer d'état. La bascule A continue toutefois de passer à l'état inverse à chaque impulsion d'entrée.

Quand ces deux entrées sont à 1, les sorties des deux portes OU sont à 1, parce qu'au moins une des sorties des portes ET qui alimentent les portes OU sera à 1, selon l'état de la bascule correspondante. Ainsi les entrées J et K des trois bascules sont à 1 et ces trois bascules commutent à l'arrivée de chaque impulsion d'entrée.

7-8 COMPTEURS PRÉRÉGLABLES

Plusieurs compteurs réalisés en circuits intégrés sont *préréglables*; on veut dire par là qu'il est possible d'installer dans le compteur, en tout temps, un nombre de départ; on peut installer ce nombre soit de façon asynchrone (indépendamment du signal d'horloge) ou de façon synchrone (au moment de la transition vraie du signal d'horloge). L'action de préréglage est aussi appelée *chargement* du compteur.

La figure 7-19 nous montre un compteur parallèle préréglable de 3 bits. Les entrées J, K et *CLK* sont câblées pour constituer un compteur parallèle. On utilise les entrées *RAU* et *RAZ* pour le prérégler de façon asynchrone. On peut installer dans le compteur n'importe quel nombre de départ, et cela à tout moment, en procédant comme suit:

1. Appliquez la valeur du nombre de départ recherché aux entrées de données parallèles, P_2, P_1 et P_0.
2. Amenez momentanément à 0 la borne CHARGEMENT PARALLÈLE, $\overline{PL}$.

Cette procédure réalisera le transfert asynchrone des niveaux sur P_2, P_1 et P_0, respectivement dans les bascules Q_2, Q_1 et Q_0 (section 5-17). Ce *transfert parallèle* survient indépendamment des entrées J, K et *CLK*. L'effet de l'entrée *CLK* demeure neutralisé tant que $\overline{PL}$ est au niveau BAS, soit dans son état vrai, étant donné que chaque bascule a une de ses entrées asynchrones actives quand $\overline{PL} = 0$. Au moment où $\overline{PL}$ revient au niveau HAUT, les bascules peuvent alors réagir aux entrées *CLK* et le comptage peut reprendre en commençant par le nombre qui a été chargé dans le compteur.

Par exemple, supposons que $P_2 = 1$, $P_1 = 0$ et $P_0 = 1$. Tant que $\overline{PL}$ est au niveau HAUT, les entrées de la donnée parallèle sont sans action. Si des impulsions d'horloge surviennent, le compteur dénombre normalement la suite de ses états. À présent, supposons que $\overline{PL}$ passe momentanément au niveau BAS au moment où la valeur 010 se trouve dans le compteur (c.-à-d. $Q_2 = 0$, $Q_1 = 1$ et $Q_0 = 0$). Ce niveau BAS sur $\overline{PL}$ induit un niveau BAS sur l'entrée RAZ de Q_1 et sur les entrées RAU de Q_2 et Q_0, de sorte que le compteur passe dans l'état 101, *quelle que soit la valeur du signal sur la borne d'horloge CLK*. Et le compteur garde l'état 101 tant que $\overline{PL}$ reste désactivé (c.-à-d. ne revient pas

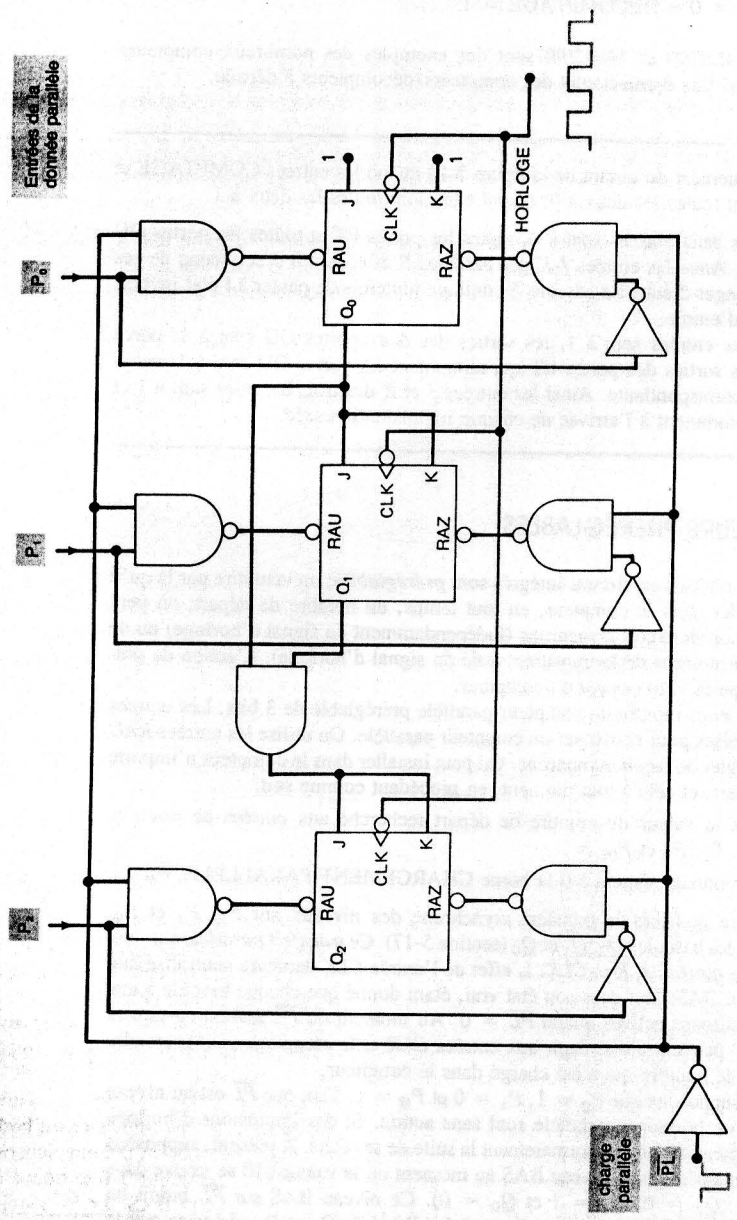


FIGURE 7-19 Compteur parallèle préréglable avec initialisation asynchrone.

au niveau HAUT); lorsqu'il y revient, le compteur reprend son dénombrement à chaque signal d'horloge, mais en partant de 101.

Le préréglage asynchrone est une solution employée dans nombre de compteurs intégrés, comme les 74190, 74191, 74192 et 74193 TTL et dans leurs équivalents CMOS, les 74HC190, 74HC191, 74HC192 et 74HC193.

Préréglage synchrone Un grand nombre de compteurs parallèles intégrés ont recours au *préréglage synchrone* à l'instant du front actif du signal qui commande également le comptage. Le niveau logique appliqué à l'entrée $\overline{PL}$ détermine si le front actif règle à une valeur initiale le compteur, ou bien s'il commande une action de comptage normale.

Parmi les exemples de puces compteurs disposant d'un préréglage synchrone, mentionnons: les TTL 74160, 74161, 74162 et 74163 et leurs équivalents CMOS, les boîtiers 74HC160, 74HC161, 74HC162 et 74HC163.

QUESTIONS DE RÉVISION

1. Que voulons-nous dire quand nous indiquons qu'un compteur est préréglable?
2. Quelle est la différence entre préréglage asynchrone et préréglage synchrone?

7-9 LE COMPTEUR 74193 (LS193/HC193)

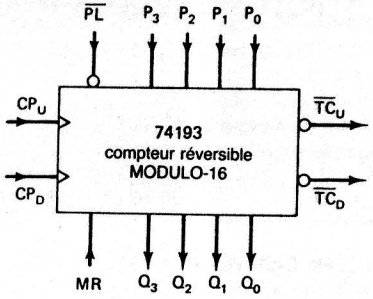
La figure 7-20 reproduit le symbole logique et décrit les entrées et les sorties du compteur 74193. On peut dire que ce compteur est un compteur/décompteur préréglable MODULO-16 fonctionnant de façon synchrone et possédant des bornes asynchrones de préréglage et d'initialisation maîtresse. Examinons plus attentivement le rôle de chacune des entrées et des sorties.

Entrées du signal d'horloge CP_U et CP_D Ce compteur réagit lors du passage des fronts montants de l'un des deux signaux d'horloge. CP_U est la borne du signal d'horloge de *comptage*. Les impulsions qui arrivent à cette entrée provoquent l'incrémentation (le comptage progressif) au moment de chaque front montant jusqu'au nombre maximal de 1111; le compteur est ensuite recyclé à 0000 et reprend le dénombrement des états. CP_D est la borne du signal d'horloge *de décomptage*. Quand des impulsions arrivent à cette entrée, le compteur décrémente (compte régressivement) au passage de chaque front montant jusqu'au nombre minimal 0000; il est ensuite recyclé à 1111 et reprend le dénombrement des états. Donc, lorsqu'une des bornes du signal d'horloge est utilisée pour le comptage, l'autre borne est invalidée (gardée au niveau HAUT).

Réinitialisation maîtresse (MR) Il s'agit ici d'une entrée asynchrone vraie au niveau HAUT qui réinitialise le compteur à l'état 0000. *MR* est une borne sur laquelle on applique une tension continue, de sorte qu'elle maintient le compteur à 0000 aussi longtemps que son niveau est à 1. Cette entrée a la priorité sur *toutes* les autres entrées.

Entrées de préréglage Il est possible d'installer dans les bascules du compteur les niveaux logiques présents sur les bornes de données parallèles P_3–P_0, en appliquant momentanément une impulsion sur la borne de chargement parallèle $\overline{PL}$ qui va de 1 à 0. Il s'agit ici d'un préréglage asynchrone qui a priorité sur l'opération de comptage. $\overline{PL}$ n'a toutefois aucun effet si l'entrée *MR* est à 1, son niveau vrai.

Sorties de comptage Le nombre affiché dans les bascules est lu sur les sorties Q_3–Q_0, où Q_0 est le bit de poids faible et Q_3, le bit de poids fort.



a)

désignations des broches	Description
CP_U	entrée du signal d'horloge de comptage (front montant actif)
CP_D	entrée du signal d'horloge de décomptage (front montant actif)
MR	entrée de réinitialisation principale asynchrone (vraie au niveau HAUT)
$\overline{PL}$	entrée de la charge parallèle asynchrone (vraie au niveau BAS)
P_0-P_3	entrées des données parallèles
Q_0-Q_3	sorties des bascules
$\overline{TC}_D$	sortie de décomptage résultante (vraie au niveau BAS)
$\overline{TC}_U$	sortie de comptage résultante (vraie au niveau BAS)

b)

sélection du mode

MR	$\overline{PL}$	CP_U	CP_D	Mode
H	X	X	X	Initial. asynchrone
B	B	X	X	Prérégl. asynchrone
B	H	H	H	Pas de changement
B	H	↑	H	Comptage
B	H	H	↑	Décomptage

H = HAUT; B = BAS
X = Indifférent; ↑ = FM

c)

FIGURE 7-20 Compteur/décompteur préréglable 74193: a) son symbole logique; b) la description de ses entrées et de ses sorties; c) table de sélection du mode. (Gracieuseté de Fairchild, filiale Schlumberger).

Sorties de comptage résultantes On utilise ces sorties lorsqu'on raccorde au moins deux puces 74193 pour construire un compteur à plusieurs étages ayant un MODULO plus grand. En mode de comptage, la sortie $\overline{TC}_U$ du compteur de rang inférieur est raccordée à l'entrée CP_U du compteur suivant de rang supérieur. En mode de décomptage, la sortie $\overline{TC}_D$ du compteur de rang inférieur est connectée à l'entrée CP_D du compteur suivant de rang supérieur.

$\overline{TC}_U$ est la sortie de *comptage résultante* (appelée également *report*). Dans la puce 74193 ce signal est produit au moyen d'un circuit logique comme celui de la figure 7-21 a). Évidemment, $\overline{TC}_U$ occupe la valeur 0 seulement lorsque le compteur est dans l'état 1111 et que CP_U est au niveau 0. Donc $\overline{TC}_U$ demeure au niveau HAUT tant que le chiffre du compteur est entre 0000 et 1110. Lorsque survient le front montant suivant sur CP_U, le nombre passe à 1111, mais $\overline{TC}_U$ ne passe pas à 0 tant que CP_U ne revient pas au niveau BAS au front montant suivant; le compteur est recyclé à 0000 ce qui amène $\overline{TC}_U$ à 1. Le front montant que l'on observe à la borne $\overline{TC}_U$ n'a lieu que lorsque le compteur est recyclé de 1111 à 0000 et on peut l'utiliser comme signal d'horloge délivré à un second compteur 74193 qui dénombre les chiffres de rangs supérieurs.

$\overline{TC}_D$ est la sortie de *décomptage résultante* (appelée également *retenue*). Elle est produite de la manière illustrée à la figure 7-21 b). Normalement elle est au niveau HAUT et ne passe au niveau BAS que lorsque le compteur a atteint l'état 0000 et que CP_D est à 0. Quand survient le front montant suivant à la borne CP_D, le compteur est recyclé à 1111, ce qui ramène $\overline{TC}_D$ au niveau HAUT. Le front montant qui apparaît ainsi à la borne $\overline{TC}_D$ peut servir d'horloge à un second décompteur 74193 qui dénombre les rangs inférieurs.

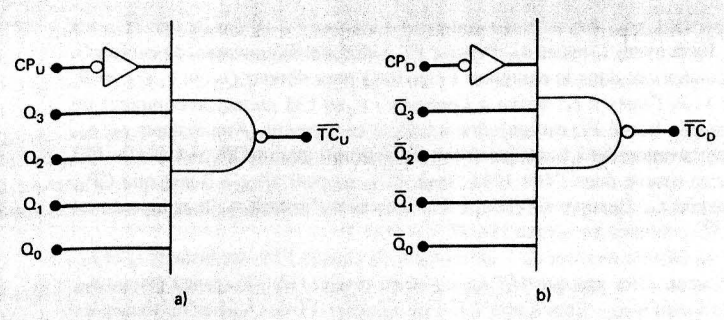
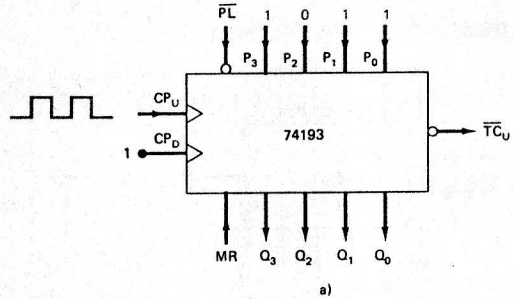


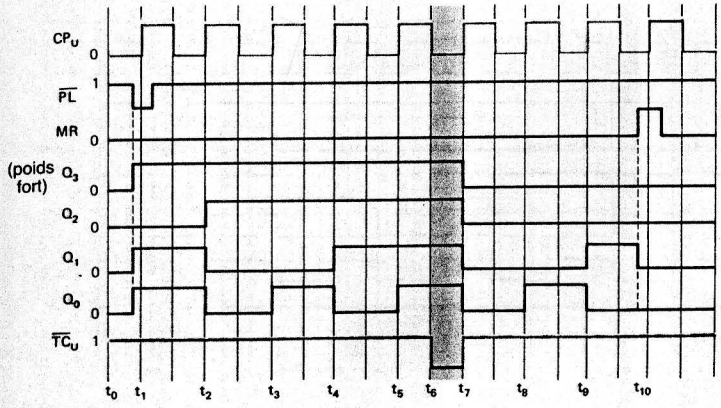
FIGURE 7-21 a) Porte logique dans 74193 qui produit $\overline{TC}_U$; b) porte logique qui produit $\overline{TC}_D$.

EXEMPLE 7-14

Reportez-vous à la figure 7-22 a), qui montre un 74193 câblé pour agir comme compteur. Les entrées de données parallèles sont constamment à l'état 1011, et les formes d'ondes des entrées CP_U, $\overline{PL}$ et MR sont illustrées à la figure 7-22 b). Supposez que l'état initial du compteur est 0000; déterminez les formes d'ondes de sortie du compteur.

FIGURE 7-22 Exemple 7-14.





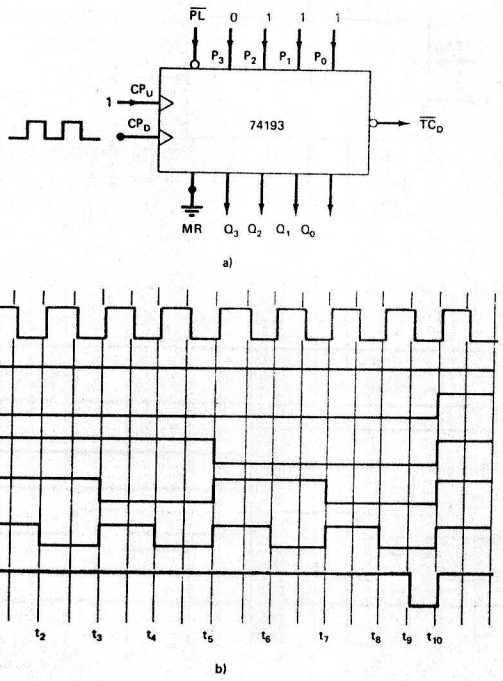
Solution Au départ (à t_0) les bascules du compteur sont toutes à 0. De ce fait $\overline{TC}_U$ est au niveau HAUT. Juste avant l'instant t_1, l'entrée $\overline{PL}$ passe momentanément au niveau 0. Ceci installe immédiatement dans le compteur l'état 1011 pour donner $Q_3 = 1$, $Q_2 = 0$, $Q_1 = 1$ et $Q_0 = 1$. À l'instant t_1, arrive à l'entrée CP_U un FM auquel le compteur ne peut réagir parce que la borne $\overline{PL}$ est toujours validée à ce moment. Aux instants t_2, t_3, t_4 et t_5, le compteur avance de 1 à chaque front montant qui atteint CP_U. Après le FM de t_5, le compteur se trouve dans l'état 1111, mais $\overline{TC}_U$ ne peut aller à 0 tant que CP_U n'est pas à 0 à l'instant t_6. Lorsque survient le FM suivant à l'instant t_7, le compteur est recyclé à 0000 et $\overline{TC}_U$ revient au niveau HAUT.

Le compteur va ensuite avancer de 1 au moment de chaque FM aux instants t_8 et t_9. Le FM de t_{10} n'a aucun effet puisque *MR* est passé au niveau HAUT avant l'instant t_{10} et reste à ce niveau jusqu'à t_{10}. Ceci a pour effet de ramener à 0 les sorties de toutes les bascules; cette action a priorité sur le signal CP_U.

EXEMPLE 7-15

On peut voir à la figure 7-23 a) le 74193 câblé de façon à réaliser un décompteur. Les sorties de données parallèles sont gardées en permanence dans l'état 0111, et les formes d'ondes de CP_D et $\overline{PL}$ sont illustrées à la figure 7-23 b). Supposez que l'état initial du compteur est 0000; déterminez les formes d'ondes de sortie.

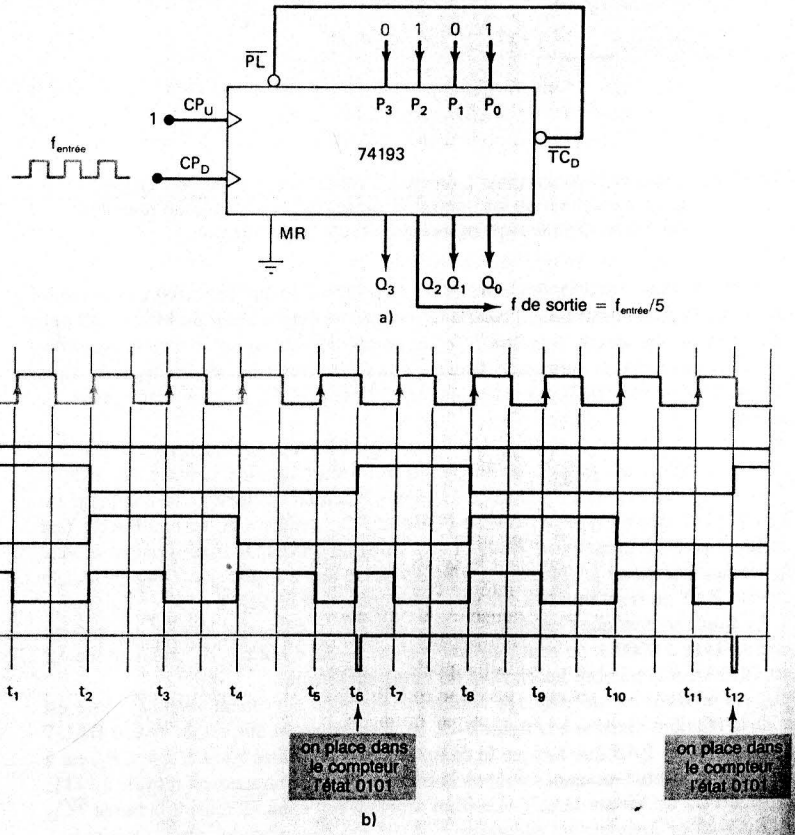
FIGURE 7-23 Exemple 7-15.



Solution À l'instant t_0 toutes les sorties des bascules sont à 0 et CP_D est aussi à 0. Ce sont là les conditions qui donnent lieu à $\overline{TC}_D = 0$. Avant l'instant t_1 la sortie $\overline{PL}$ passe momentanément à 0. Ceci installe immédiatement dans le compteur l'état 0111 et fait passer la borne $\overline{TC}_D$ à la valeur 1. Le FM qui arrive à CP_D à l'instant t_1 n'a aucun effet, puisque $\overline{PL}$ est toujours effectif. Le compteur réagit aux FM des instants t_2 à t_8 et compte régressivement jusqu'à l'état 0000, atteint en t_8. $\overline{TC}_D$ ne revient pas à 0 avant t_9, lorsque CP_D passe à 0. À l'instant t_{10}, le front montant appliqué à CP_D provoque le recyclage du compteur à 1111 et ramène $\overline{TC}_D$ à la valeur 1.

Fabrication de compteurs à MODULO variable au moyen du 74193 Il est facile de câbler des compteurs préréglables pour obtenir différents MODULO sans devoir ajouter des circuits logiques supplémentaires. C'est ce que nous allons démontrer dans le cas du 74193 en étudiant le circuit de la figure 7-24 a). Dans cet exemple, le 74193 est utilisé

FIGURE 7-24 a) La puce 74193 câblée pour réaliser un compteur MODULO-5; b) formes d'ondes.



comme un décompteur dont les entrées de données parallèles sont gardées en permanence dans l'état 0101 (5_{10}). Remarquez que la sortie $\overline{TC}_D$ est ramenée à l'entrée $\overline{PL}$.

Commençons notre analyse en supposant que ce circuit est en train de compter régressivement et se trouve à l'instant t_0 dans l'état 0101. Reportez-vous à la figure 7-24 b) pour voir les formes d'ondes du compteur.

Ce compteur décrémente (compte régressivement) au moment du passage des FM de CP_D, soit aux instants t_1 à t_5. À l'instant t_5, le compteur se trouve dans l'état 0000. Quand CP_D passe à 0 au moment t_6, il force $\overline{TC}_D$ à passer aussi à 0. Ceci a pour effet de valider immédiatement l'entrée $\overline{PL}$ et de replacer dans le compteur l'état 0101. Remarquez que $\overline{TC}_D$ demeure au niveau BAS pendant un court instant seulement, puisque les sorties du compteur en se plaçant dans l'état 0101 en réponse à $\overline{PL} = 0$ provoquent le retrait de la condition $\overline{TC}_D = 0$. C'est ce qui explique le signal transitoire étroit illustré pour $\overline{TC}_D$.

La même suite d'opérations se répète pendant l'intervalle t_7 à t_{12} et les intervalles suivants de même durée. Si on examine la forme d'onde Q_2, on voit qu'elle effectue un cycle complet chaque fois que CP_D effectue *cinq* cycles. Par exemple, il y a *cinq* cycles du signal d'horloge entre le FM de Q_2 à l'instant t_6 et le FM de Q_2 à l'instant t_{11}. Donc la fréquence de la forme d'onde Q_2 est égale à un cinquième de la fréquence du signal d'horloge.

Ce n'est pas une coïncidence si le rapport des fréquences (5) est le même que le nombre appliqué aux entrées des données parallèles (0101 = 5). En fait, on peut changer la division de la fréquence en changeant les niveaux logiques appliqués aux entrées de données parallèles.

On réalise donc facilement un *diviseur de fréquence variable* en connectant des interrupteurs aux entrées des données parallèles du circuit de la figure 7-24. Ces interrupteurs peuvent être positionnés pour donner la valeur correspondant au MODULO recherché.

QUESTIONS DE RÉVISION

1. Décrivez le rôle de $\overline{PL}$ et des entrées P_0–P_3.
2. Décrivez le rôle de l'entrée *MR*.
3. *Vrai ou faux:* On ne peut forcer une valeur dans 74193 lorsque *MR* est à son niveau vrai.
4. Dites quels niveaux logiques doivent se trouver sur les entrées CP_D, $\overline{PL}$ et *MR* pour que le 74193 compte les impulsions qui arrivent à CP_U.

7-10 COMPLÉMENT À LA NOTATION FONCTIONNELLE IEEE/ANSI*

Il est possible d'en apprendre davantage sur la notation fonctionnelle de la symbologie IEEE/ANSI en examinant le symbole IEEE/ANSI du CI 74193, illustré à la figure 7-25. Chaque type de circuit intégré que nous étudions de cette manière vient compléter votre compréhension des mécanismes de cette nouvelle symbologie et vous prépare pour l'avenir, puisque ces symboles sont appelés à être utilisés de plus en plus fréquemment.

De nouveau, on doit rappeler que seuls les libellés à l'intérieur du périmètre du rectangle sont prescrits par la norme IEEE/ANSI. Les désignations ou indications à l'extérieur du périmètre ne sont pas normalisées et peuvent varier d'un fabricant de circuits intégrés à l'autre.

* Cette section peut être omise sans rien enlever à la continuité de la présentation.

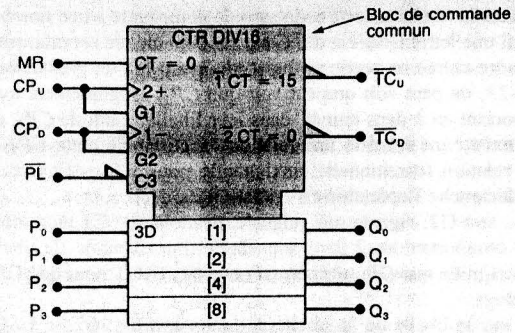


FIGURE 7-25 Symbole IEEE/ANSI du CI 74193.

Une partie des indications utilisées dans la figure 7-25 devrait déjà vous être familière. Par exemple, le périmètre du symbole se divise en deux parties: le bloc de commande commun qui agit sur toutes les bascules du compteur, et les quatre petits rectangles du bas qui figurent les bascules. Le nombre entre crochets dans chacun des rectangles désigne le poids relatif de la bascule dans le compteur. Le libellé CRT DIV 16 stipule que cet élément, quand il fonctionne normalement, est un compteur (CTR) ayant 16 états (c.-à-d. que c'est un compteur qui divise par 16). L'entrée *MR* du bloc de commande commun est associée au libellé $CT = 0$, ce qui signifie que le compteur est remis à zéro quand *MR* passe au niveau HAUT.

Fonction de commande (C) La lettre C dans le libellé associé à une entrée signale qu'il s'agit d'une entrée qui *commande* l'introduction de données dans l'élément de mémoire (soit une bascule). Généralement, C est employée pour les entrées d'horloge qui autorisent le transfert des données dans une bascule à l'instant de leur front actif. Nous avons vu cela lors de notre étude des symboles IEEE/ANSI des bascules au chapitre 5. Sur la figure 7-25, C est associée à l'entrée $\overline{PL}$, dénommée chargement parallèle, étant donné que cette entrée contrôle la pénétration des données dans les quatre bascules du compteur. Plus précisément, le libellé C3 nous informe que cette entrée contrôle toute entrée ayant le suffixe 3 dans son libellé associé. Dans le cas présent, cela englobe les entrées P_0, P_1, P_2 et P_3, puisqu'elles sont toutes associées au libellé 3D (que l'on ne voit que dans le bloc de la bascule supérieure, mais qui doit s'entendre pour toutes les autres bascules). Le « D » du libellé correspond à « donnée ».

Tout cela signifie que lorsque $\overline{PL}$ est à son niveau actif (BAS), la donnée sur P_0 à P_3 est transférée dans les bascules Q_0 à Q_3. Comme il n'y a pas le symbole du déclenchement par un signal sur la borne $\overline{PL}$, il faut comprendre que la borne $\overline{PL}$ agit aussi longtemps qu'elle demeure dans son état vrai.

Sens du comptage (+ ou −) Les entrées CP_U et CP_D apparaissent sur cette figure associées chacune à deux libellés différents, puisqu'elles causent des comportements distincts dans le compteur. D'abord, examinons le libellé du haut de l'entrée CP_U, soit 2+. Le signe plus (+) indique qu'à l'instant d'un FM de cette entrée le compteur est incrémenté, en d'autres mots, qu'il dénombre progressivement ses états. Même chose pour le libellé du haut de CP_D, 1−, où le signe moins (−) signale que le compteur est décrémenté, en d'autres mots, qu'il dénombre régressivement ses états. La signification des chiffres dans ces deux libellés sera dévoilée plus loin.

Fonction ET (G) Le lettre G dans le libellé associé à une entrée se rapporte à une fonction ET. Il faut donc interpréter ainsi une lettre G suivie d'un chiffre: cette entrée est multipliée logiquement (ET) avec toute autre entrée ou sortie ayant le même chiffre en préfixe dans sa désignation. Sur la figure 7-25, on peut voir que CP_U est multipliée logiquement avec les entrées ou les sorties comportant un 1 dans leur libellé. Le libellé du haut de CP_D est 1−, il faut en conclure qu'il existe une relation fonctionnelle de type ET entre CP_U et CP_D. Plus précisément, cette relation fonctionnelle nous informe que CP_U doit être au niveau HAUT pour que CP_D déclenche l'opération de décomptage.

Le libellé du bas de CP_D, soit G2, signale une relation fonctionnelle ET entre cette borne et toute entrée ou sortie comportant un 2 dans son libellé. Par exemple, le libellé du haut de CP_U est 2+, ce qui signifie que CP_D doit être au niveau HAUT pour que CP_U déclenche l'opération du comptage.

Penchons-nous à présent sur le libellé de la sortie $\overline{TC}_D$. Il s'agit de $\overline{2}CT = 0$. On voit un 2 dans ce libellé, ce qui indique une relation fonctionnelle ET avec CP_D. En fait, puisque qu'il s'agit d'un $\overline{2}$, la relation fonctionnelle ET est avec $\overline{CP}_D$. Ainsi, le libellé de $\overline{TC}_D$ nous apprend que cette borne passe à son niveau vrai BAS quand CP_D est BAS et que le nombre dans le compteur est zéro ($CT = 0$). La même analyse du libellé de $\overline{TC}_U$ nous apprendra que cette borne passe dans son état vrai BAS quand CP_U est BAS et que le nombre dans le compteur est 15 ($CT = 15$).

QUESTIONS DE RÉVISION

1. Dites qu'elle est la signification des expressions: fonction de commande et fonction ET.
2. Dites qu'elle est la signification des libellés suivants associés à des entrées: a) +; b) G4; c) C5; d) 5D.

7-11 DÉCODAGE D'UN COMPTEUR

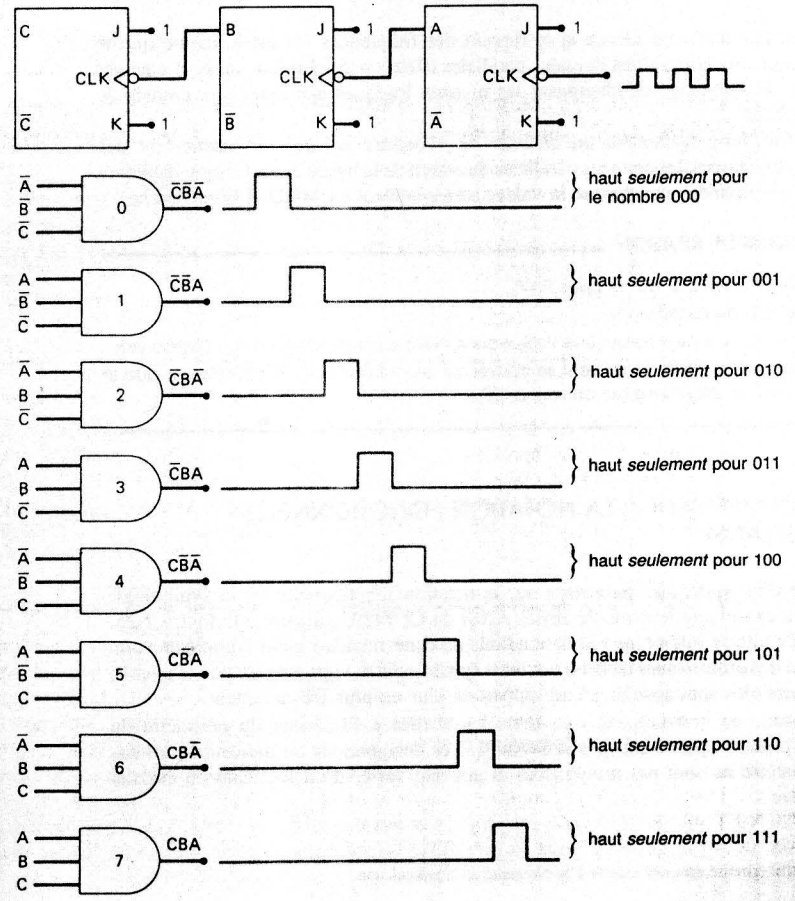
Souvent les compteurs numériques sont utilisés dans des applications où le nombre représenté par les états des bascules doit être connu ou affiché. Une des façons les plus simples d'afficher le contenu d'un compteur consiste à raccorder la sortie de chaque bascule à une petite lampe témoin. Ainsi, les états des bascules sont traduits par de petites lampes (allumée signifie 1, éteinte signifie 0); on peut trouver le nombre affiché en décodant les états binaires représentés par les lampes. Par exemple, supposons que l'on utilise cette méthode dans le cas d'un compteur DCB et que les lampes sont respectivement éteintes-allumées-allumées-éteintes. Cette succession représente l'état 0110; on sait que cet état est l'équivalent binaire du chiffre décimal 6. Les autres combinaisons des états des lampes correspondent aux autres nombres possibles.

La méthode des lampes témoins devient rapidement malcommode lorsqu'on augmente le nombre de bits du compteur, parce qu'il n'est pas facile de décoder rapidement le résultat affiché. C'est la raison pour laquelle il est préférable d'imaginer un moyen *électronique* pour décoder le contenu d'un compteur puis pour afficher les résultats selon une forme que l'on interprète immédiatement et qui ne demande pas de conversion mentale.

Toutefois, le décodage électronique d'un compteur est essentiel pour une autre raison encore plus importante: dans de nombreuses applications les compteurs servent à commander le cadencement ou l'ordonnancement automatique d'opérations sans l'intervention de l'homme. Par exemple, l'opération d'un système particulier est déclenchée lorsqu'un compteur atteint l'état 101100 (44_{10}). On peut recourir à un circuit logique pour décoder, ou détecter ce nombre particulier et lancer l'opération. Dans un système numérique, il y a beaucoup d'opérations qui sont commandées de cette manière. Évidemment, dans ce genre de procédé l'intervention de l'homme n'est pas souhaitable sauf dans les systèmes extrêmement lents.

Décodeur vrai au niveau HAUT Un compteur MODULO-X possède X états différents; chaque état est un agencement particulier de zéros et de uns mémorisés dans les bascules du compteur. Un réseau de décodage est un circuit logique qui produit X sorties différentes, chacune signalant la présence d'un état particulier du compteur. Les sorties d'un décodeur sont conçues pour produire soit un niveau HAUT ou un niveau BAS lorsqu'un état est détecté. Un décodeur vrai au niveau HAUT indique une détection en ayant un niveau HAUT sur une de ses sorties. La figure 7-26 nous fait voir le circuit de décodage complet d'un compteur MODULO-8 qui envoie un 1 pour indiquer une détection. Ce décodeur est

FIGURE 7-26 Utilisation de portes ET pour décoder un compteur MODULO-8



formé de huit portes ET à trois entrées. Chaque porte ET produit une sortie au niveau HAUT quand paraît à ses entrées un état particulier du compteur.

Par exemple, la porte ET 0 a ses entrées raccordées aux sorties des bascules $\overline{C}$, $\overline{B}$ et $\overline{A}$. La sortie de cette porte sera à 0 en tout temps sauf lorsque $A = B = C = 0$, c'est-à-dire lorsque le nombre dans le compteur est 000 (zéro). De la même façon, les entrées de la porte ET 5 sont raccordées aux sorties des bascules C, $\overline{B}$ et A, de sorte que sa sortie passe au niveau HAUT seulement quand $C = 1$, $B = 0$ et $A = 1$, c'est-à-dire quand le nombre dans le compteur est 101 (chiffre décimal 5). Les autres portes ET fonctionnent de la même manière pour les autres nombres possibles. À un instant quelconque, seulement une des sorties des portes ET est au niveau HAUT, celle qui décode le nombre particulier présent dans le compteur. Les formes d'ondes reproduites à la figure 7-26 illustrent clairement ce fonctionnement.

Les huit sorties des portes ET peuvent servir à allumer huit lampes témoins distinctes, qui correspondent aux nombres décimaux 0 à 7. À un moment donné, une seule lampe s'allume pour signaler le nombre se trouvant dans le compteur.

Le décodeur à portes ET est une solution pouvant être appliquée à des compteurs ayant un nombre quelconque d'états. Voici une illustration de ceci.

EXEMPLE 7-16

Dites combien il faut de portes ET pour décoder entièrement tous les états d'un compteur binaire MODULO-32. Dites quelles entrées sont amenées à la porte qui décode le nombre 21.

Solution Un compteur MODULO-32 a 32 états possibles. Comme il faut une porte ET pour décoder chaque état, il faut donc construire un décodeur ayant 32 portes ET. Puisque $32 = 2^5$, le compteur doit avoir 5 bascules. Donc chaque porte aura 5 entrées, une pour chaque bascule. Pour décoder le nombre 21 qui s'écrit en binaire 10101_2, il faut amener à la porte ET en question les entrées E, $\overline{D}$, C, $\overline{B}$ et A, où E est la bascule de poids le plus fort.

Décodeur vrai au niveau BAS Si on substitue aux portes ET des portes NON-ET, les sorties du décodeur seront normalement au niveau HAUT et passeront au niveau BAS seulement dans le cas du nombre décodé. Dans la pratique, on retrouve les deux types de décodeurs; on utilise celui qui convient aux circuits qui doivent être pilotés par les sorties du décodeur.

EXEMPLE 7-17

La figure 7-27 représente une situation fréquente: l'utilisation d'un compteur pour produire une forme d'onde de commande qui peut être appliquée à un moteur, à une soupape électromagnétique ou à un appareil de chauffage. Le compteur MODULO-16 compte jusqu'à 15, est recyclé, compte de nouveau jusqu'à 15 et cela indéfiniment. Chaque fois que le compteur atteint le nombre 8 (1000), la porte NON-ET supérieure voit sa sortie passée à 0, ce qui a pour effet de mettre à 1 la bascule X. Cette bascule reste au niveau HAUT jusqu'à ce que le compteur atteigne le nombre 14 (1110); à ce moment, la porte NON-ET inférieure décode ces chiffres et passe à 0 afin de mettre à 0 la bascule X. On voit donc que la sortie X est au niveau HAUT entre les nombres 8 et 14 de chaque cycle du compteur.

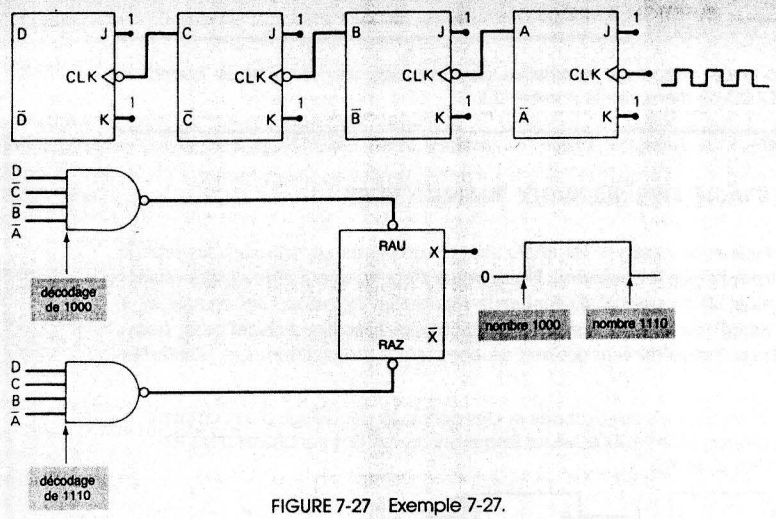
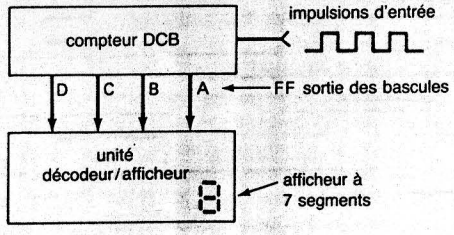


FIGURE 7-27 Exemple 7-27.

Décodage d'un compteur DCB Un compteur DCB a 10 états, que l'on peut décoder au moyen des techniques déjà décrites. Les compteurs DCB ont 10 sorties, une pour chaque chiffre décimal 0 à 9 représenté par les états des bascules du compteur. On peut utiliser ces sorties pour commander 10 lampes témoins et avoir ainsi un affichage du nombre. Toutefois, il est plus fréquent d'utiliser un seul afficheur pour représenter tous les nombres décimaux de 0 à 9. Un tel dispositif, appelé ***tube nixie*** contient 10 petits filaments, épousant la forme des chiffres, empilés les uns par-dessus les autres. Les sorties du décodeur DCB contrôlent ces filaments. Il existe une autre catégorie d'afficheurs décimaux qui contiennent 7 petits segments faits d'un matériau qui émet de la lumière lorsqu'il est traversé par un courant électrique (généralement des filaments incandescents ou des diodes électroluminescentes). Les sorties du décodeur DCB déterminent quels segments sont éclairés afin de produire la forme représentant un des chiffres décimaux.

Nous verrons plus en détails les genres de décodeurs et d'afficheurs au chapitre 9. Toutefois, étant donné que les compteurs DCB, ainsi que les décodeurs et afficheurs qui les accompagnent, sont très courants, c'est l'unité décodeur/afficheur (voyez la figure 7-28) que nous avons choisie pour représenter les circuits complets qui permettent d'afficher sous forme d'un chiffre décimal le contenu d'un compteur DCB.

FIGURE 7-28 Le contenu des compteurs DCB est généralement représenté par un seul dispositif d'affichage.



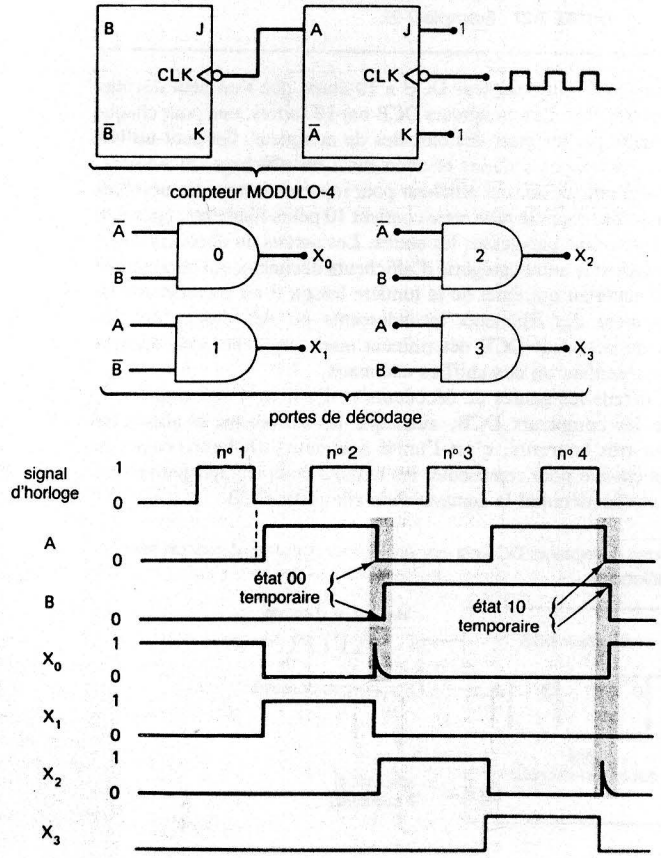
QUESTION DE RÉVISION

1. Décrivez la porte de décodage nécessaire pour produire une sortie basse quand un compteur MODULO-60 mémorise le nombre 23.

7-12 DÉCODAGE DES SIGNAUX TRANSITOIRES

À la section 7-5, nous avons analysé les effets sur les compteurs asynchrones des retards de propagation introduits par les bascules. Nous avons alors vu que l'addition des retards de propagation a pour effet essentiel de limiter la réponse en fréquence des compteurs à propagation. Les retards entre les transitions associées aux bascules peuvent aussi occasionner des problèmes lorsqu'on veut décoder un compteur à propagation. Les problèmes dont nous voulons parler sont des signaux transitoires ou des décrochements à la sortie de certaines des portes de décodage. On illustre ceci à la figure 7-29 pour le cas d'un compteur à propagation MODULO-4.

FIGURE 7-29 Formes d'ondes des bascules et des portes de décodage d'un compteur à propagation MODULO-4 sur lesquelles on voit des transitoires dans les sorties X_0 et X_2.



Les formes d'ondes apparaissant aux sorties des bascules et des portes de décodage sont illustrées sur cette figure. Remarquez le retard de propagation entre la forme d'onde du signal d'horloge et la forme d'onde de la sortie A, et entre la forme d'onde A et la forme d'onde B. Les transitoires que l'on observe dans les formes d'ondes de décodage X_0 et X_2 sont dus au retard entre les formes d'ondes A et B. X_0 est la sortie de la porte ET qui décode le nombre 00. La condition 00 se produit aussi momentanément lorsque le compteur passe de 01 à 10, comme on peut le voir sur les formes d'ondes. Il en est ainsi parce que B ne peut changer d'état avant que A passe au niveau BAS. Ce passage momentané à l'état 00 ne dure que quelques nanosecodes (suivant le t_{pd} de la bascule B); cependant, cette condition peut être décodée si la réponse de la porte est suffisamment rapide. D'où le décrochement observé à la sortie X_0.

Une situation tout à fait analogue produit un transitoire à la sortie X_2. X_2 est la sortie de décodage de la condition 10, et cette condition se produit momentanément quand le compteur passe de l'état 11 à l'état 00 au moment de la quatrième impulsion d'horloge, comme on peut le voir sur les formes d'ondes. De nouveau, la cause de ceci est le retard de la réponse de la bascule B après le passage de la bascule A au niveau 0.

Cette situation, que nous venons de décrire pour un compteur MODULO-4, est vraie pour n'importe quel compteur à propagation. Ce genre de situation survient dans les compteurs à propagation à cause de leur principe de fonctionnement, c'est-à-dire celui de la réaction en chaîne: chaque bascule déclenche la suivante et ainsi de suite. Les décrochements observés aux sorties des décodeurs peuvent être ou non un problème, selon l'application que l'on fait du compteur. Quand on utilise le compteur pour compter des impulsions et afficher des résultats, les décrochements des sorties de décodage ne portent pas à conséquence parce que leur durée est très courte et que l'afficheur ne réagit pas à leur présence. Toutefois, quand le compteur sert à commander d'autres circuits logiques, comme celui de la figure 7-27, les décrochements peuvent entraîner un fonctionnement inapproprié. Par exemple, à la figure 7-27 un décrochement à la sortie de l'une ou l'autre des portes NON-ET de décodage aurait mis à 1 ou à 0 la bascule X au mauvais moment.

Nous parvenons à prédire quand dans la séquence d'un compteur asynchrone survient un état temporaire en parcourant tous les états d'un compteur, une bascule à la fois. Par exemple, examinons pas à pas le processus réel, dans un compteur à propagation, du passage de l'état 011 (3) à l'état 100 (4):

	C	B	A		
	0	1	1	(3)	
États temporaires	0	1	0	(2)	⟵ La bascule A commute en premier
	0	0	0	(0)	⟵ et provoque le basculement de B
	1	0	0	(4)	⟵ qui entraîne le basculement de C

Notez l'apparition de deux états temporaires: 010 et 000.

Dans les cas où il est impossible de fonctionner avec les décrochements des sorties de décodage, on a recours à deux solutions de base. La première consiste à utiliser un compteur parallèle plutôt qu'un compteur à propagation. Rappelons que dans un compteur parallèle les bascules sont toutes déclenchées au même moment par les impulsions d'horloge, de sorte que les conditions qui donnent naissance au décrochement n'existent pas. Toutefois, même dans un compteur parallèle, il peut y avoir des décrochements parce que les bascules n'ont pas toutes le même t_{pd}, particulièrement lorsque certaines bascules alimentent une charge beaucoup plus élevée que celle des autres.

Strobage Une méthode beaucoup plus fiable pour éliminer les décrochements sur les sorties des décodeurs est celle appelée *strobage*. Dans cette technique on utilise un signal appelé *signal de strobage* pour invalider les portes ET de décodage (garder leurs sorties à 0) jusqu'à ce que toutes les bascules soient dans un état stable après le front descendant du signal d'horloge. La figure 7-30 a) montre comment fonctionne cette technique: un signal de strobage est appliqué en entrée à chaque porte de décodage. Les formes d'ondes illustrées montrent que le signal de strobage passe au niveau BAS quand l'impulsion d'horloge passe au niveau HAUT. Tant que le signal de strobage est au niveau BAS, les sorties des portes de décodage sont aussi gardées au niveau BAS. Le signal strobe passe au niveau HAUT un certain temps t_D *après* que l'impulsion d'horloge a passée à 0 en vue de valider les portes. t_D est choisi de façon à être plus grand que le temps total que prend le compteur pour atteindre un état stable; ce temps dépend donc évidemment des retards introduits par les bascules et du nombre de bascules dans le compteur. Grâce à cette technique, il n'y aura aucun décrochement sur les sorties des portes qui sont invalidées pendant que les sorties des bascules sont instables.

La méthode du strobage n'est pas retenue si un compteur sert seulement à l'affichage, puisque les décrochements apparaissant sur les sorties des portes sont trop étroits pour perturber les afficheurs. On a recours à un signal de strobage quand le compteur sert dans des applications de commande comme celle de la figure 7-27, là où des décrochements peuvent entraîner un fonctionnement erroné.

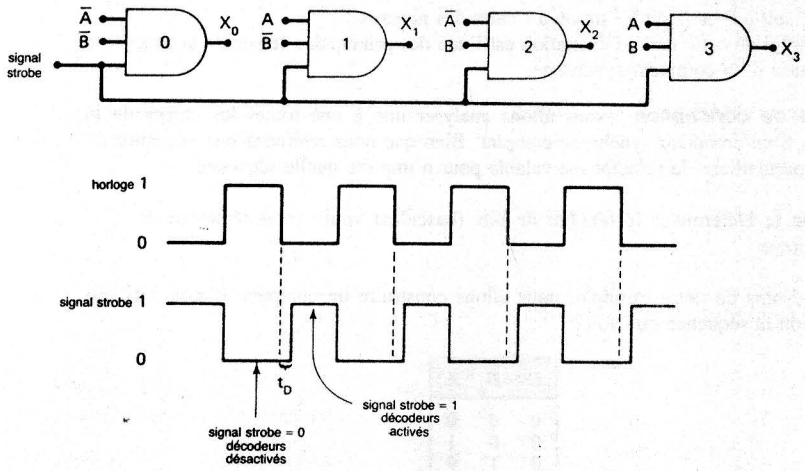


FIGURE 7-30 Utilisation d'un signal strobe pour éliminer les décrochements des portes de décodage.

QUESTIONS DE RÉVISION

1. Dites comment il se fait que des transitoires peuvent apparaître sur les sorties des portes de décodage d'un compteur asynchrone.
2. Dites comment le strobage élimine les transitoires des portes de décodage.

7-13 COMPTEURS DCB MONTÉS EN CASCADE

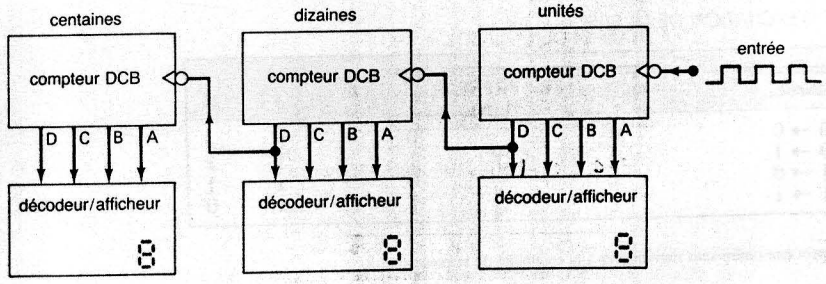
Chaque fois qu'on doit compter les impulsions et afficher le résultat en numération décimale, on utilise des compteurs DCB. Un seul compteur DCB peut compter de 0 à 9 et ensuite être recyclé à 0. Pour compter des valeurs décimales plus grandes, on peut monter en cascades des compteurs DCB comme à la figure 7-31. Le fonctionnement de ce montage est le suivant:

1. Au début tous les compteurs sont initialisés à 0. Donc le nombre décimal affiché est 000.
2. À mesure qu'arrivent des impulsions, le compteur DCB des unités augmente de 1 à chaque impulsion. Après 9 impulsions, les compteurs DCB des dizaines et des centaines sont toujours à 0 et le compteur des unités est à 9 (nombre binaire 1001). Donc, le nombre décimal affiché est 009.
3. À la dixième impulsion d'entrée, le compteur des unités est recyclé à 0, entraînant le passage de 1 à 0 de la sortie de la bascule *D*. Cette transition agit comme signal d'horloge pour le compteur des dizaines qui est incrémenté. Ainsi après 10 impulsions, la valeur décimale affichée est 010.
4. À mesure qu'arrivent d'autres impulsions, le compteur des unités est incrémenté à chaque impulsion et chaque fois qu'il est recyclé à 0, le compteur des dizaines augmente de 1. Donc après 99 impulsions, le compteur des dizaines est à 9 tout comme celui des unités. Le nombre décimal affiché est 099.
5. À la centième impulsion, le compteur des unités est recyclé à 0 ce qui entraîne également le recyclage du compteur des dizaines à 0. La sortie de la bascule *D* du compteur des dizaines effectue une transition de 1 à 0 qui agit comme signal d'horloge pour le compteur des centaines et provoque son incrémentation. Ainsi après 100 impulsions, on lit le chiffre décimal 100.
6. Ce processus se poursuit jusqu'à la 999e impulsion. À la 1000e impulsion, tous les compteurs sont recyclés à 0.

On voit facilement qu'un tel montage peut être modifié pour représenter le nombre de chiffres décimaux que nous voulons en ajoutant de nouveaux étages. Par exemple, pour compter jusqu'à 999 999, il nous faut six compteurs DCB avec leur décodeur et leur afficheur. En général, il nous faut un compteur DCB par chiffre décimal.

À titre d'information, mentionnons que les compteurs DCB de la figure 7-31 peuvent être soit des puces 74293 câblées pour réaliser des compteurs MODULO-10, ou des compteurs DCB intégrés comme le 7490 ou le 74192.

FIGURE 7-31 Compteurs DCB montés en cascades en vue de compter et d'afficher les nombres de 000 à 999.



7-14 CONCEPTION D'UN COMPTEUR SYNCHRONE*

On retrouve sous forme de circuits intégrés différents types de compteurs — asynchrones, synchrones et combinés asynchrones/synchrones. La majorité de ces dispositifs comptent la suite normale des nombres binaires, bien qu'il soit possible de modifier la séquence de comptage en recourant à des solutions comme celles mises en pratique pour les CI 74293 et 74193. Il se produit des cas où il nous faut un compteur dont la séquence des états ne correspond pas à la suite normale des nombres binaires, comme 000, 010, 101, 001, 000,...

On peut mettre en application plusieurs méthodes pour concevoir des compteurs qui parcourent une séquence arbitraire. Nous nous proposons d'exposer les détails d'une méthode courante dans laquelle on monte en compteur synchrone des bascules J-K. La même méthode peut être étendue aux bascules D. La technique présentée ici est la procédure de conception des circuits numériques, appelée **conception de systèmes séquentiels**, que l'on enseigne à un niveau supérieur.

Principe de base Dans les compteurs synchrones, toutes les bascules sont déclenchées par l'horloge au même instant. Avant chaque impulsion d'horloge, chacune des entrées J et K des bascules du compteur doit se trouve dans le niveau approprié qui assure le passage de chaque bascule dans le bon état. Par exemple, considérons le cas suivant:

État présent			État suivant		
C	B	A	C	B	A
1	0	1	0	1	1

À l'instant du signal d'horloge suivant, les entrées J et K des bascules doivent se trouver aux bons niveaux pour que la bascule C passe de 1 à 0, que la bascule B passe de 0 à 1 et que la bascule A passe de 1 à 1 (aucun changement).

Le processus de synthèse d'un compteur synchrone se résume alors à la conception des circuits logiques qui *décodent* les divers états du compteur pour les délivrer à chacune des entrées J et K. Les bornes d'entrée de ce décodeur coïncident avec les sorties d'une ou de plusieurs bascules. Un exemple de ceci est le compteur synchrone de la figure 7-17 où la porte ET, qui alimente les entrées J et K de la bascule C, décode les états des bascules A et B. De la même façon, la porte ET, qui alimente les entrées J et K de la bascule D, décode les états de A, B et C.

Table des états d'excitation J-K Avant de commencer la conception proprement dite du décodeur de chaque entrée J et K, nous allons revoir le fonctionnement de la bascule J-K, mais cette fois-ci en partant d'une approche différente: celle de la *table d'excitation* illustrée ci-dessous.

TABLE D'EXCITATION DE LA BASCULE J-K

Transition en sortie	État PRÉSENT Q (N)	État SUIVANT Q (N + 1)	J	K
0 → 0	0	0	0	x
0 → 1	0	1	1	x
1 → 0	1	0	x	1
1 → 1	1	1	x	0

* Cette section peut être omise sans rien enlever à la continuité de la présentation.

La colonne de gauche de cette table présente les transitions possibles de la sortie de la bascule. Les deuxième et troisième colonnes contiennent, pour chacune des transitions, l'état présent de la bascule, noté Q (N), et l'état suivant, noté Q (N + 1). La dernière colonne contient les niveaux J et K à chacune des transitions. Examinons chacune de ces situations.

TRANSITION 0 → 0 L'état présent de la bascule est 0 et y demeure à l'arrivée du signal d'horloge. D'après ce que nous savons du fonctionnement de la bascule J-K, on déduit que cela se produit quand $J = K = 0$ (condition pas de changement) ou quand $J = 0$ et $K = 1$ (condition de remise à zéro). Donc, il faut que $J = 0$ alors que K peut prendre l'une ou l'autre valeur. La table nous informe de ceci par un « 0 » sous J et par un « x » sous K. Rappelons qu'un « x » indique la condition « indifférente ».

TRANSITION 0 → 1 L'état présent est 0, et la sortie de la bascule change pour 1. C'est ce qui se passe quand $J = 1$ et $K = 0$ (condition remise à un) ou quand $J = K = 1$ (condition de basculement). Donc, cette transition survient quand $J = 1$ et que K est à l'un ou l'autre des niveaux.

TRANSITION 1 → 0 L'état présent est 1, et la sortie de la bascule change pour 0. Cela se produit quand $J = 0$ et $K = 1$ ou quand $J = K = 1$. Cette transition se réalise si K est à 1 et que J est à l'un ou l'autre des niveaux.

TRANSITION 1 → 1 L'état présent est 1 et y demeure à l'arrivée du signal d'horloge. Cette situation se réalise quand $J = K = 0$ ou quand $J = 1$ et $K = 0$. Donc, il faut que K soit à 0 et J soit à l'un ou à l'autre des niveaux.

L'emploi de cette table d'excitation est l'une des principales étapes de la procédure de conception d'un compteur synchrone.

Procédure de conception Nous allons analyser une à une toutes les étapes de la conception d'un compteur synchrone complet. Bien que nous retenions une séquence de comptage particulière, le principe est valable pour n'importe quelle séquence.

Étape 1: Déterminer le nombre de bits (bascules) voulu et la séquence de comptage.

Pour les besoins de notre exemple, nous allons construire un compteur à trois bits qui compte selon la séquence que voici:

C	B	A
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
0	0	0
etc.		

Notez l'absence dans cette séquence des états 101, 110 et 111. Nous allons désigner ces états par le nom *d'états indésirables*.

Étape 2: Tracer le graphe des états pour **tous** les états possibles, y compris ceux qui ne figurent pas dans la séquence de comptage voulue.

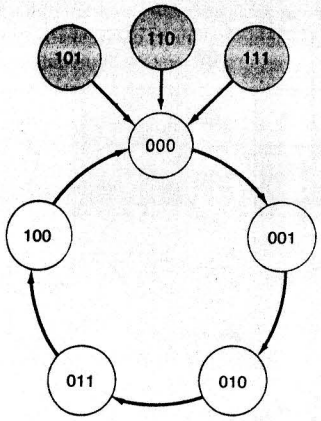


FIGURE 7-32 Graphe des états pour notre exemple de conception d'un compteur synchrone.

Pour notre exemple, le graphe des états se présente comme sur la figure 7-32. Les états 000 à 100 se suivent comme dans la suite normale. La nouveauté dans ce graphe est d'inclure les états indésirables. Il est nécessaire d'en tenir compte dans notre conception au cas où un de ceux-ci apparaîtrait dans le compteur à la mise sous tension ou en raison du bruit. Le concepteur de circuits est libre de choisir l'état dans lequel passe le compteur au signal d'horloge suivant après ces états indésirables. Pour notre part, nous avons choisi de ramener le compteur dans l'état 000, le début de la séquence.

Étape 3: À partir du graphe des états, construire une table énumérant **tous** les états PRÉSENTS et leurs états SUIVANTS.

La table résultante de cette dernière étape est:

	état PRÉSENT			état SUIVANT		
	C	*B*	*A*	*C*	*B*	*A*
Ligne 1	0	0	0	0	0	1
2	0	0	1	0	1	0
3	0	1	0	0	1	1
4	0	1	1	1	0	0
5	1	0	0	0	0	0
6	1	0	1	0	0	0
7	1	1	0	0	0	0
8	1	1	1	0	0	0

La partie gauche de la table renferme *tous* les états possibles, même ceux qui ne figurent pas dans la séquence. Nous appelons ceux-ci états PRÉSENTS. La partie droite de la table énumère l'état suivant correspondant à chaque état PRÉSENT. Ces états suivants sont déduits du graphe des états de la figure 7-32. Par exemple, la ligne 1 nous fait voir qu'à l'état PRÉSENT 000 correspond l'état SUIVANT 001, et la ligne 5, qu'à l'état PRÉSENT

100 fait suite l'état 000. Les lignes 6, 7 et 8 nous apprennent que l'état SUIVANT des états PRÉSENTS indésirables 101, 110 et 111 est 000.

Étape 4: Additionner à cette table une colonne pour chacune des entrées *J* et *K*. Pour chaque état PRÉSENT, indiquer les niveaux qui doivent se trouver sur les entrées *J* et *K* pour que la transition se fasse vers l'état SUIVANT.

Notre exemple de conception a recours à trois bascules — *C*, *B* et *A* — ayant chacune des entrées *J* et *K*. Par conséquent, c'est six colonnes qu'il faut ajouter dans la table. Voici la table complète appelée **table d'excitation du circuit**.

TABLE D'EXCITATION DU CIRCUIT

	état PRÉSENT			état SUIVANT								
	C	*B*	*A*	*C*	*B*	*A*	J_C	K_C	J_B	K_B	J_A	K_A
Ligne 1	0	0	0	0	0	1	0	*x*	0	*x*	1	*x*
2	0	0	1	0	1	0	0	*x*	1	*x*	*x*	1
3	0	1	0	0	1	1	0	*x*	*x*	0	1	*x*
4	0	1	1	1	0	0	1	*x*	*x*	1	*x*	1
5	1	0	0	0	0	0	*x*	1	0	*x*	0	*x*
6	1	0	1	0	0	0	*x*	1	0	*x*	*x*	1
7	1	1	0	0	0	0	*x*	1	*x*	1	0	*x*
8	1	1	1	0	0	0	*x*	1	*x*	1	*x*	1

Ces six nouvelles colonnes sont les entrées *J* et *K* de chacune des bascules. Les valeurs attribuées à ces entrées ont été trouvées au moyen de la table d'excitation élaborée précédemment. Nous allons démontrer la procédure suivie pour plusieurs cas; nous vous demandons de vérifier les autres cas.

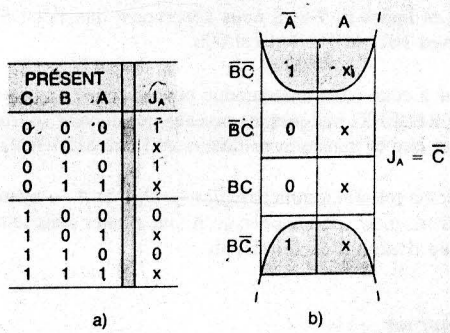
Commençons par la ligne 1. L'état PRÉSENT est 000; le compteur passe à l'état SUIVANT 001 à l'arrivée d'un signal d'horloge. Pour que cette transition s'effectue, il faut que la bascule *C* transite de 0 à 0. D'après la table d'excitation d'un J-K, il faut que $J_C = 0$ et que K_C soit « *x* ». En plus, il faut que la bascule *B* passe de 0 à 0, ce qui se produit quand $J_B = 0$ et $K_B =$ « *x* ». Pour sa part, la bascule *A* effectue la transition 1 à 0, et d'après la table d'excitation, cela a lieu quand $J_A = 1$ et $K_A = x$.

Maintenant, analysons la ligne 4: l'état PRÉSENT est 011 et l'état SUIVANT EST 100. Pour que cette transition d'état réussise, la bascule *C* doit passer de 0 à 1, ce qui se produit quand $J_C = 1$ et $K_C = x$. Les bascules *B* et *A* transitent, toutes les deux, de 1 à 0. La table d'excitation nous apprend que pour cela il faut, pour ces deux bascules, que $J = x$ et $K = 1$.

Pour toutes les autres lignes, on détermine de la même manière les valeurs qui doivent se trouver sur *J* et *K* pour que s'effectue la transition.

Étape 5: Concever les circuits logiques qui produisent les niveaux pour chacune des entrées *J* et *K*.

La table d'excitation du circuit renferme six entrées *J* et *K* — J_C, K_C, J_B, K_B, J_A et K_A. Nous devons considérer que chacune est la sortie d'un circuit logique dont les entrées proviennent des bascules *C*, *B* et *A*. Nous devons donc, comme prochaine étape, concevoir chacun de ces circuits. Faisons ce travail pour le circuit de J_A.



PRÉSENT C	B	A	J_A
0	0	0	1
0	0	1	x
0	1	0	1
0	1	1	x
1	0	0	0
1	0	1	x
1	1	0	0
1	1	1	x

FIGURE 7-33 a) Partie de la table d'excitation montrant le comportement de J_A pour chaque état PRÉSENT; b) diagramme K utilisé pour simplifier l'expression de J_A.

Pour cela, nous devons trouver, pour chaque cas, les états PRÉSENTS de C, B et A, de même que les niveaux recherchés en J_A. Ces renseignements ont été tirés de la table d'excitation et résumés dans la table de la figure 7-33 a). Cette table de vérité nous fait connaître les niveaux exigés sur J_A pour chaque état PRÉSENT. Évidemment, pour certains des cas, la valeur sur J_A est « indifférente ». Pour élaborer le circuit logique propre à J_A, il est nécessaire de trouver l'expression de J_A en fonction de C, B et A. C'est ce que nous faisons en transposant le contenu de la table de vérité dans un diagramme de Karnaugh et en procédant à la simplification, comme à la figure 7-33 b).

Il n'y a que deux 1 dans ce diagramme de Karnaugh, que nous pouvons réunir afin de construire le terme $\overline{A}\overline{C}$; de plus si nous choisissons 1 pour x dans la case $A\overline{B}\overline{C}$ et la case $AB\overline{C}$, nous parvenons à réunir un quartet et à tout réduire au simple terme de $\overline{C}$. Donc, l'expression finale est:

$$J_A = \overline{C}$$

À présent, étudions le cas de K_A. Nous suivons la même démarche que pour J_A.

Cependant, un examen des valeurs de la table d'excitation pour K_A démontre qu'il n'y a que des 1 et des cas indifférents. En remplaçant les x par des 1, on constate que K_A est toujours au niveau 1. L'expression finale pour cette sortie est dès lors:

$$K_A = 1$$

En raisonnant de la même façon, on parvient à dériver des expressions pour J_C, K_C, J_B et K_B. Les diagrammes K de ces expressions sont illustrés à la figure 7-34. Vous pouvez en vérifier l'exactitude en vous référant à la table d'excitation du circuit.

Étape 6: Matérialiser les expressions finales.

Les circuits de décodage des entrées J et K sont réalisés à partir des expressions déduites des diagrammes K. Le dessin du compteur synchrone complet est illustré à la figure 7-35. Remarquez que toutes les bascules sont commandées en même temps par le signal d'horloge. Vous pouvez vérifier que les entrées J et K ont le comportement logique recherché.

Dernières remarques Nous venons d'étudier la conception complète d'un compteur synchrone construit avec des bascules J-K. La même démarche peut être mise en pratique pour un compteur ayant un nombre de bits quelconque et suivant une séquence quelconque.

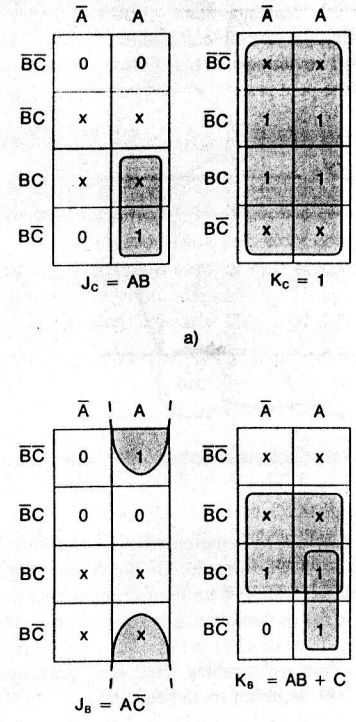
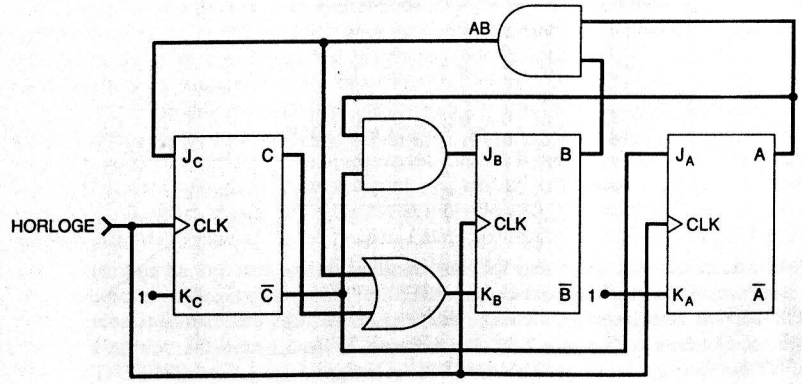


FIGURE 7-34 a) Diagrammes de Karnaugh des circuits logiques de J_C et K_C; diagrammes de Karnaugh des circuits logiques de J_B et K_B.

FIGURE 7-35 Réalisation finale de notre exemple de conception d'un compteur synchrone.



La seule restriction imposée à la séquence étant qu'on ne doit pas y trouver deux fois la même valeur durant un cycle. Par exemple, la séquence 000, 001, 011, 001, 100 est inadmissible puisque l'état 001 revient deux fois. Cette restriction est indiscutable puisque notre méthode de conception ne peut prendre en considération un état PRÉSENT qui aurait deux états SUIVANTS.

QUESTIONS DE RÉVISION

1. Énumérez les six étapes de la conception d'un compteur synchrone.
2. Quelle information fournit une table d'excitation des J-K?
3. Quelle information est contenue dans la table d'excitation du circuit?
4. *Vrai ou faux:* La méthode de conception d'un compteur synchrone peut être appliquée à la séquence suivante: 0010, 0011, 0100, 0111, 1010, 1110, 1111.

7-15 COMPTEURS CIRCULAIRES

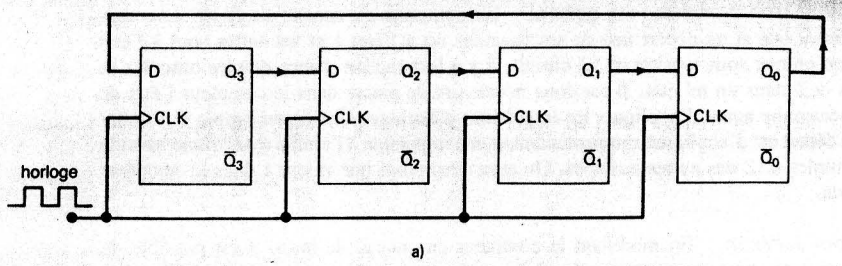
Il est possible d'utiliser des registres à décalage pour réaliser différents types de compteurs. Tous les *compteurs circulaires* ont recours à la *rétroaction*, c'est-à-dire au rétrocouplage de la sortie de la dernière bascule du registre à décalage sur l'entrée de la première bascule. Les compteurs à décalage les plus courants sont le compteur en anneau et le compteur Johnson.

Compteur en anneau Le compteur circulaire le plus élémentaire qui soit est un registre à décalage **circulant** dans lequel la valeur de la dernière bascule est ramenée à l'entrée de la première bascule. On peut voir un tel compteur à la figure 7-36 réalisé au moyen de bascules de type D (on aurait pu utiliser des bascules J-K). Les bascules sont connectées de façon à ce que l'information soit décalée de gauche à droite et ramenée de Q_0 à Q_3. Généralement, il n'y a qu'un 1 dans le registre, et il circule aussi longtemps que des impulsions d'horloge sont appliquées aux bornes *CLK*. C'est la raison pour laquelle on l'appelle *compteur en anneau*.

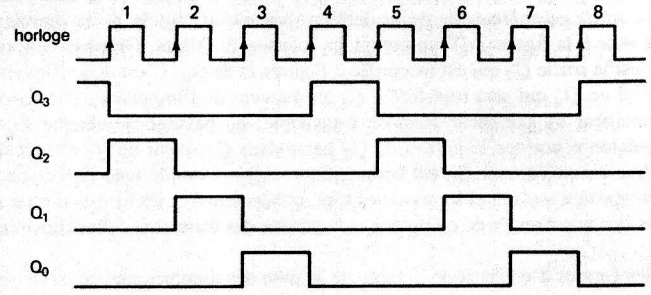
Les formes d'ondes, la table de la suite des nombres, et le graphe des états reproduits à la figure 7-36, illustrent les divers états des bascules à mesure qu'arrivent des impulsions; dans cette figure l'état initial est $Q_3 = 1$ et $Q_2 = Q_1 = Q_0 = 0$. Après la première impulsion, le 1 a été décalé de Q_3 à Q_2 de sorte que le compteur se retrouve dans l'état 0100. La seconde impulsion amène le compteur dans l'état 0010 et la troisième impulsion le place dans l'état 0001. À la *quatrième* impulsion, le 1 qui se trouvait à la borne Q_0 est ramené à Q_3, ce qui réinstalle dans le compteur l'état 1000, évidemment l'état initial. La même séquence d'événements est répétée pour les impulsions ultérieures.

Ce compteur fonctionne comme s'il avait un MODULO-4, puisqu'il possède quatre états distincts par cycle. Même si la suite des nombres dans ce circuit ne correspond pas à la suite normale des nombres binaires, on dit que c'est un compteur parce que chaque nombre correspond à un état particulier des bascules. Notez que la fréquence de chaque forme d'onde de sortie d'une bascule est égale à un quart de la fréquence du signal d'horloge, car il s'agit ici d'un compteur en anneau MODULO-4.

On peut construire des compteurs en anneau ayant n'importe quel MODULO; pour obtenir un compteur en anneau MODULO-*N*, il faut monter *N* bascules comme sur la figure 7-36. En règle générale, un compteur en anneau a besoin de plus de bascules, pour le même MODULO, qu'un compteur binaire; par exemple, un compteur en anneau



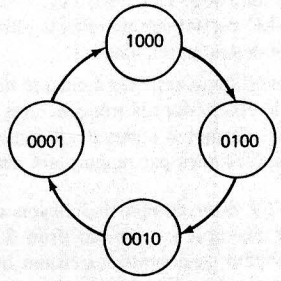
a)



b)

Q_3	Q_2	Q_1	Q_0	impulsion d'horloge
1	0	0	0	0
0	1	0	0	1
0	0	1	0	2
0	0	0	1	3
1	0	0	0	4
0	1	0	0	5
0	0	1	0	6
0	0	0	1	7
⋮	⋮	⋮	⋮	⋮

c)



d)

FIGURE 7-36 a) Compteur en anneau de 4 bits; b) formes d'ondes; c) table de la suite des nombres.

MODULO-8 nécessite 8 bascules tandis qu'un compteur binaire MODULO-8 n'en a besoin que de 3.

Même s'il est vrai qu'un compteur en anneau utilise davantage de bascules, c'est un circuit utile parce qu'il peut être décodé sans l'utilisation de portes de décodage. Le signal de décodage de chaque état est prélevé à la sortie de sa bascule correspondante. Comparez les formes d'ondes des bascules du compteur en anneau à celles des portes de décodage de la figure 7-26. Dans certains cas, il se pourra que le compteur en anneau constitue une meilleure solution que le compteur binaire et ses portes de décodage associées, particulièrement dans des applications où le compteur sert à commander l'ordonnancement des opérations d'un système.

Lancement d'un compteur en anneau Un compteur en anneau ne fonctionne convenablement que si au départ une de ses bascules est à l'état 1 et les autres sont à l'état 0. Lorsqu'on met sous tension un tel circuit, il y a fort peu de chance que les bascules se trouvent déjà dans un tel état. Il est donc nécessaire de placer dans le compteur l'état de départ nécessaire avant d'appliquer les impulsions d'horloge. Une façon simple d'installer l'état de départ est d'appliquer momentanément une impulsion à l'entrée *RAU* d'une bascule et aux entrées *RAZ* des autres bascules. On aura ainsi placé une valeur 1 dans le compteur en anneau.

Compteur Johnson En modifiant le compteur en anneau de base, il est possible de réaliser un autre type de compteur circulaire, présentant certaines propriétés différentes. On construit le compteur *Johnson* exactement comme le compteur en anneau normal, sauf qu'on raccorde la sortie *complémentée* de la dernière bascule à l'entrée de la première bascule. On peut voir à la figure 7-37 un compteur Johnson de 3 bits. On peut voir sur cette figure que c'est la sortie $\overline{Q}_0$ qui est raccordée à l'entrée D de Q_2. C'est donc l'inverse du niveau mémorisé en Q_0 qui sera transféré à Q_3 au moment de l'impulsion d'horloge.

Le fonctionnement du compteur Johnson est simple: au passage de chaque front montant de l'impulsion d'horloge le niveau de Q_2 passe dans Q_1, celui de Q_1 est décalé dans Q_0 et l'inverse du niveau sur Q_0 est communiqué à Q_2. Compte tenu de ce fonctionnement et en supposant que toutes les bascules sont initialement à 0, les formes d'ondes, la table de la suite des nombres de ce compteur et le graphe des états sont celles illustrées à la figure 7-37.

L'examen des formes d'ondes et de la table de la suite des nombres met en évidence les aspects suivants:

1. Le compteur prend les six états suivants avant que ne se répète la séquence d'opération: 000, 100, 110, 111, 011 et 001. C'est donc un compteur Johnson MODULO-6. Remarquez que la suite des états ne correspond pas à la suite normale des nombres binaires.
2. La forme d'onde associée à chaque bascule est une onde carrée (facteur d'utilisation de 50 %) dont la fréquence est un sixième de la fréquence d'horloge. En outre, les formes d'ondes des bascules sont décalées d'une période du signal d'horloge les unes par rapport aux autres.

Le MODULO d'un compteur Johnson est toujours égal à *deux fois* le nombre de bascules. Par exemple, si on raccorde 5 bascules, comme dans le circuit de la figure 7-37, on obtient un compteur Johnson MODULO-10 dans lequel la forme d'onde de chaque sortie d'une bascule est une onde carrée dont la fréquence est un dixième de la fréquence d'horloge. Pour construire un compteur Johnson MODULO-N (où N est un nombre pair) on utilise $N/2$ bascules.

Décodage d'un compteur Johnson Pour un MODULO donné, un compteur Johnson a besoin de deux fois moins de bascules qu'un compteur en anneau. Toutefois pour décoder un compteur Johnson, il faut des portes alors qu'on n'en a pas besoin pour décoder un compteur en anneau. Le compteur Johnson, comme le compteur binaire, a besoin d'une porte logique pour chaque nombre à décoder; cependant, dans le cas du compteur Johnson, les portes auront toujours deux entrées quel que soit le nombre de bascules dans le compteur. On peut voir à la figure 7-38 les portes de décodage pour les six états du compteur Johnson de la figure 7-37.

Remarquez que chaque porte de décodage possède seulement deux entrées, même s'il y a trois bascules dans le compteur. Ceci s'explique du fait que pour chaque nombre il y a toujours deux des trois bascules qui forment une combinaison unique dans la suite

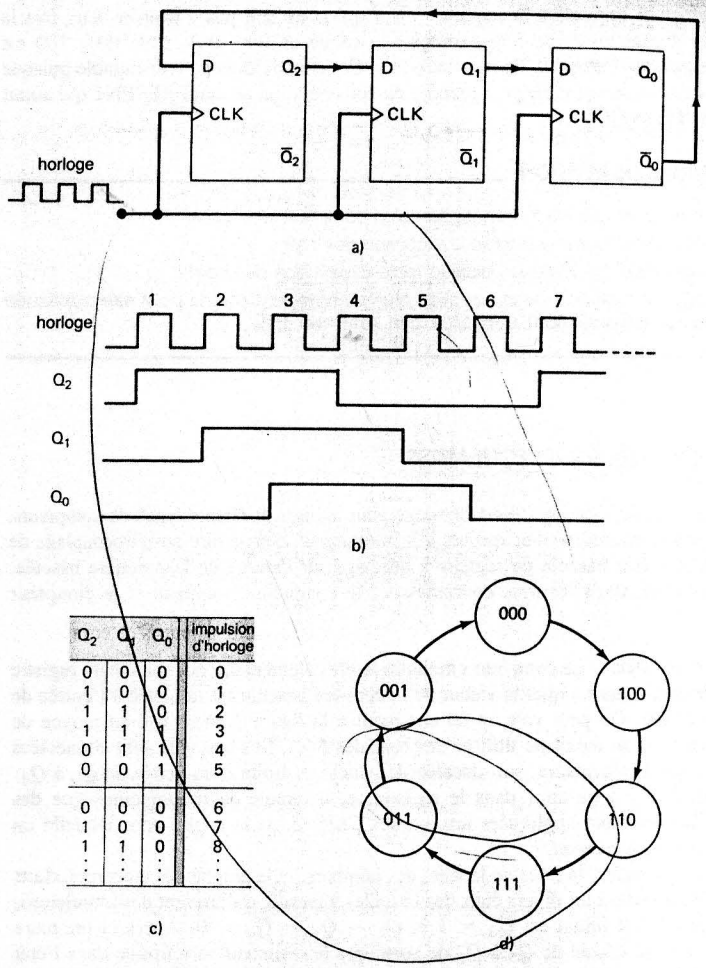


FIGURE 7-37 a) Compteur Johnson MODULO-6; b) formes d'ondes; c) table de la suite des nombres.

des états. Par exemple, la combinaison $Q_2 = Q_0 = 0$ ne se retrouve qu'une fois dans la suite des nombres, c'est-à-dire quand le nombre est 0. Donc la porte ET 0 dont les entrées sont $\overline{Q}_2$ et $\overline{Q}_0$ sert à décoder ce nombre. Cette caractéristique est commune à tous les autres états de la suite, comme le lecteur pourra le vérifier. En fait, pour tous les compteurs Johnson, on n'a besoin que de portes de décodage à deux entrées.

Les compteurs Johnson sont des solutions intermédiaires entre les compteurs en anneau et les compteurs binaires. Un compteur Johnson a besoin d'un moins grand nombre de bascules que le compteur en anneau, mais habituellement d'un plus grand nombre que le compteur binaire. Contrairement au compteur en anneau, on doit lui associer des circuits de décodage qui sont toutefois moins complexes que ceux du compteur binaire. Finalement, ce compteur peut dans certaines situations s'avérer un choix logique.

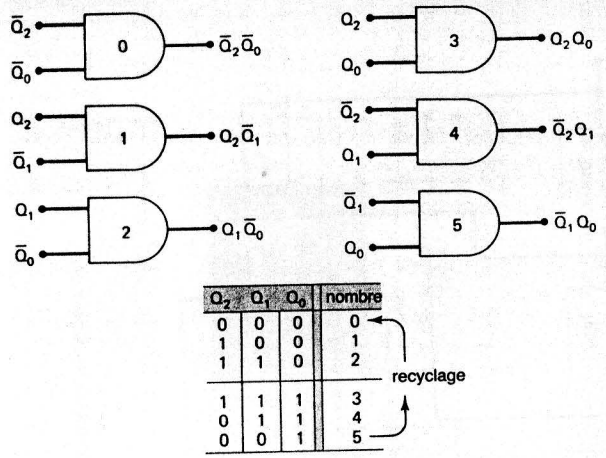


FIGURE 7-38 Circuit de décodage d'un compteur Johnson MODULO-6.

Compteurs circulaires intégrés Il n'y a pas beaucoup de compteurs circulaires ou de compteurs Johnson réalisés en circuits intégrés. La raison en est relativement simple, il est très facile de prendre un registre à décalage CI et de le câbler pour réaliser un compteur en anneau ou un compteur Johnson. Il existe toutefois quelques puces de compteur Johnson CMOS (74HC4017, 74HC4022) dans lesquelles ont été intégrés les circuits complets de décodage.

QUESTIONS DE RÉVISION

1. Dites lequel des compteurs circulaires nécessite le plus de bascules pour un MODULO donné.
2. Dites lequel des compteurs circulaires nécessite le plus de circuits de décodage.
3. Montrez comment transformer un compteur en anneau en un compteur Johnson.
4. *Vrai ou faux:*
   a) Les sorties d'un compteur en anneau sont toujours des ondes carrées.
   b) Les circuits de décodage d'un compteur Johnson sont moins complexes que ceux d'un compteur binaire.
   c) Le compteur en anneau et le compteur Johnson sont des compteurs synchrones.

7-16 APPLICATIONS: COMPTEUR DE FRÉQUENCE

Il existe beaucoup d'applications pour les types de compteurs que nous venons d'étudier. Dans la présente section et la suivante, nous décrivons deux applications des compteurs dans les systèmes numériques.

Un *compteur de fréquence* (ou *fréquencemètre*) est un circuit qui mesure et affiche la fréquence d'un signal impulsionnel. L'une des méthodes les plus faciles pour construire un compteur de fréquence est illustrée schématiquement sur la figure 7-39 a). On peut voir sur cette figure un compteur, ses circuits décodeurs/afficheurs associés et une porte ET.

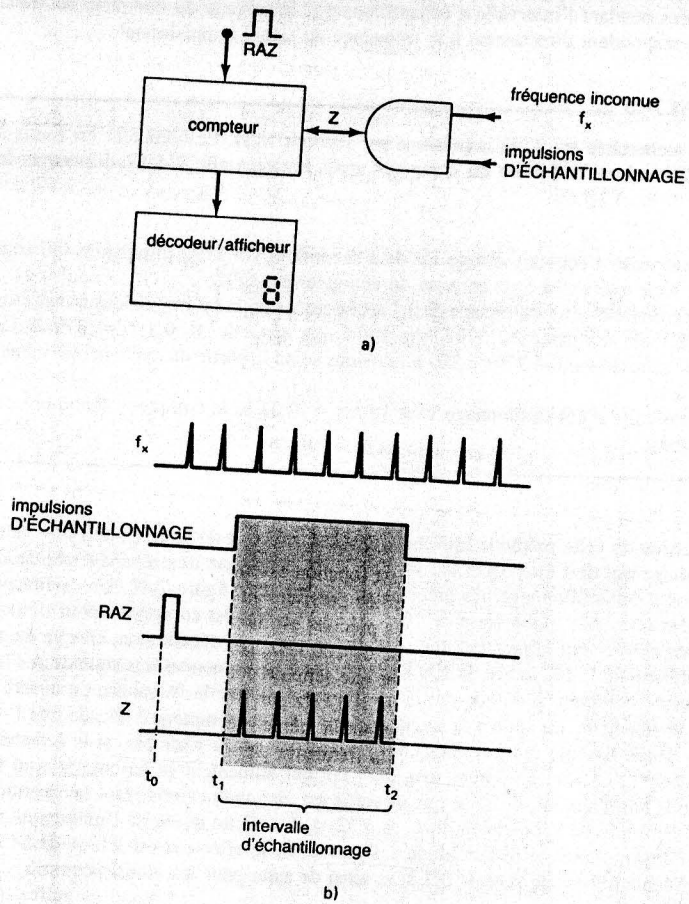


FIGURE 7-39 Principe d'un compteur de fréquence.

Les entrées de la porte ET sont d'une part le signal de fréquence inconnue, f_x, et une impulsion D'ÉCHANTILLONNAGE qui contrôle la durée pendant laquelle les impulsions sont transmises à travers la porte ET jusqu'au compteur. Ce compteur est un montage en cascade de compteurs DCB (figure 7-31), chacun ayant une unité décodeur/afficheur qui convertit les sorties DCB en un affichage décimal facile à lire.

Les formes d'ondes de la figure 7-39 b) montrent que l'impulsion *RAZ* est appliquée au compteur à l'instant t_0 pour initialiser le compteur. Avant t_1, l'impulsion D'ÉCHANTILLONNAGE est au niveau BAS, ce qui met la sortie de la porte ET, *Z*, au niveau BAS et n'envoie aucune impulsion au compteur. L'impulsion D'ÉCHANTILLONNAGE est au niveau HAUT entre t_1 et t_2: on appelle ce laps de temps **l'intervalle d'échantillonnage**. Durant l'intervalle d'échantillonnage, les impulsions de fréquence inconnue traversent la porte ET et sont comptées. Après t_2, la sortie de la porte ET revient à 0 et le compteur

cesse de dénombrer les impulsions. Donc, le compteur aura mémorisé le nombre d'impulsions arrivées pendant l'intervalle d'échantillonnage; le contenu du compteur est donc un nombre correspondant directement à la fréquence du signal impulsionnel.

EXEMPLE 7-18

La fréquence recherchée est 3792 impulsions par seconde (i/s). Le compteur est remis à 0 avant t_1. Déterminez la lecture du compteur après un intervalle d'échantillonnage de a) 1 s, b) 0,1 s, et c) 10 ms.

Solution

a) Quand l'intervalle d'échantillonnage est de 1 seconde il y a 3792 impulsions qui sont comptées, de sorte que après t_2 le contenu du compteur est 3792.

b) Avec un intervalle d'échantillonnage de 0,1 seconde, le nombre d'impulsions transmises par la porte ET au compteur est 3792 impulsions par seconde $\times$ 0,1 s = 379,2. Le compteur aura donc dénombré 379 ou 380 impulsions selon la partie du cycle impulsionnel qui est passé à t_1.

c) Quand l'intervalle d'échantillonnage vaut 10 ms = 0,01 s, le compteur dénombre 37 ou 38 impulsions.

La précision de cette méthode est fonction essentiellement de la durée de l'intervalle d'échantillonnage qui doit être contrôlée précisément. Une façon très répandue d'obtenir des impulsions d'échantillonnage très précises est montrée à la figure 7-40. Un oscillateur à quartz génère une onde très précise de 100 kHz; cette onde est conformée pour donner des impulsions carrées qui alimentent une série de compteurs à décade dont chaque étage divise successivement la fréquence de 100 kHz par 10. Les fréquences des signaux sur les sorties de chaque compteur à décade ont la même précision que la fréquence du quartz.

Grâce au sélecteur, on choisit la fréquence de sortie du compteur à décade que l'on veut délivrer à une bascule qui introduit une division par 2. Par exemple, si le sélecteur est à la position 1, ce sont les impulsions de 1 Hz qui alimentent la bascule Q, qui se trouve en mode basculement, de sorte que sa sortie est une onde carrée ayant une période $T = 2$ s et une durée de l'impulsion de $t_p = T/2 = 1$ s. Cette durée de l'impulsion est l'intervalle d'échantillonnage recherché de 1 seconde. Si le sélecteur est à la position 2, l'intervalle d'échantillonnage sera de 0,1 s; et ainsi de suite pour les autres positions.

EXEMPLE 7-19

Présumez que le compteur de la figure 7-39 est formé de trois compteurs DCB montés en cascades et des afficheurs associés. Si la fréquence inconnue se situe entre 1 et 9,99 kHz, dites sur quelle position on doit mettre le sélecteur de la figure 7-40.

Solution Le nombre maximal que l'on peut compter avec 3 compteurs DCB est 999. Une fréquence de 9,99 kHz place le nombre 999 dans le compteur si l'intervalle d'échantillonnage est de 0,1 s. Donc pour exploiter toutes les capacités du compteur, on doit placer le sélecteur à la position 2. Si on avait utilisé un intervalle d'échantillonnage d'une seconde, pour la gamme de fréquences précisée, le compteur aurait pu dépasser son nombre maximal. Si on avait choisi un intervalle plus court, le contenu du compteur aurait été compris entre 0 et 99; ceci donne des lectures ayant seulement deux chiffres significatifs. D'autant plus que l'on n'aurait pas exploité toutes les possibilités du compteur.

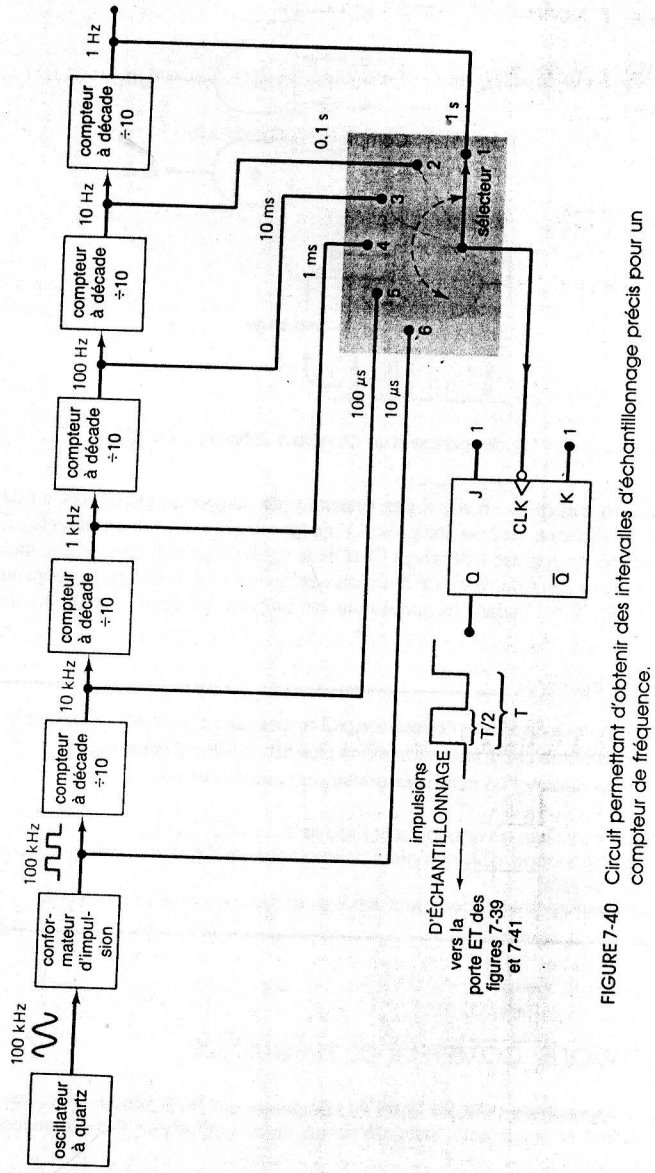
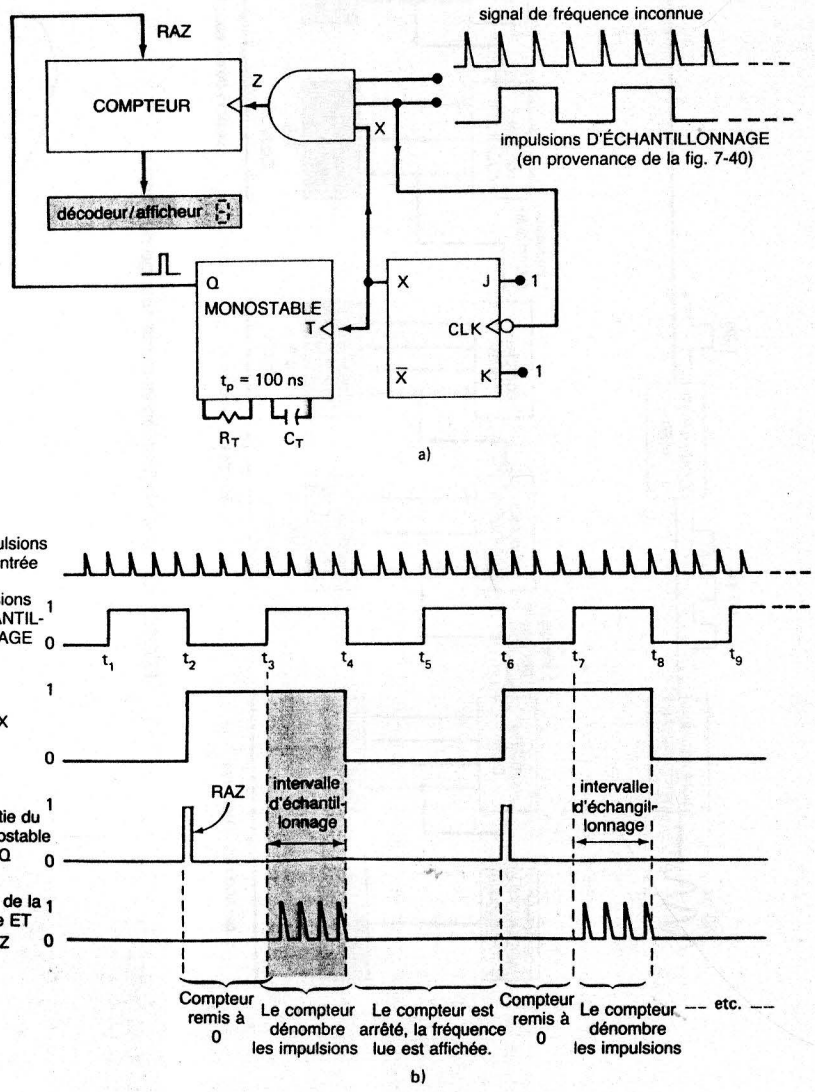


FIGURE 7-40 Circuit permettant d'obtenir des intervalles d'échantillonnage précis pour un compteur de fréquence.

Compteur de fréquence complet Examinons maintenant le compteur de fréquence plus complet de la figure 7-41 a). Ce circuit, contrairement à l'autre, contient un monostable et une bascule J-K fonctionnant en mode de basculement de même qu'une porte ET à trois entrées, la troisième entrée accueillant la sortie X de la bascule. Les impulsions D'ÉCHANTILLONNAGE sont appliquées à la porte ET ainsi qu'à l'entrée *CLK* de la bascule. De

FIGURE 7-41 Compteur de fréquence.



telles impulsions sont générées par un circuit comme celui de la figure 7-40. La marche à suivre détaillée que voici se réfère aux formes d'ondes de la figure 7-41 b).

1. Supposez que la bascule X est à l'état 0 (elle est passée à 0 au moment du front descendant de l'impulsion d'échantillonnage précédente).
2. Cette valeur BASSE sur X est appliquée à l'entrée de la porte ET, ce qui invalide sa sortie; aucune impulsion n'est communiquée au compteur même si la première impulsion d'échantillonnage se produit entre t_1 et t_2.
3. À l'instant t_2, le FD de la première impulsion d'échantillonnage met la bascule X dans l'état opposé 1 (notez que $J = K = 1$). La transition de 0 à 1 apparaissant à la borne X déclenche le monostable qui produit une impulsion de 100 ns pour *remettre à zéro* le compteur. La valeur affichée par le compteur est maintenant *zéro*.
4. À l'instant t_3, la seconde impulsion d'échantillonnage valide la porte ET (puisque maintenant X est à 1) et laisse passer les impulsions de fréquence inconnue jusqu'au compteur pour qu'il les dénombre jusqu'à t_4.
5. À l'instant t_4, l'impulsion d'échantillonnage revient à 0 ce qui provoque le basculement de X à la valeur 0, et la porte ET se retrouve invalidée. Le compteur cesse de dénombrer les impulsions.
6. Entre t_4 et t_6, le compteur conserve et affiche le nombre atteint à l'instant t_4. Notez que la troisième impulsion d'échantillonnage ne valide pas la porte ET parce que la sortie X de la bascule est à 0.
7. À l'instant t_6, le FD de l'impulsion d'échantillonnage provoque le basculement de X au niveau 1 et la séquence des opérations décrites depuis t_2 reprend.

Ce compteur de fréquence s'engage alors dans une suite répétitive de remise à zéro, de comptage, d'attente de l'affichage, de remise à zéro, de comptage... Par exemple, supposons un compteur à trois étages DCB et afficheur de trois bits. Pour un intervalle d'échantillonnage d'une seconde (1 s) et une fréquence inconnue de 237 ips, le compteur et l'unité d'affichage répètent indéfiniment la séquence que voici:

- Mise à zéro et affichage de zéro pendant 1 s [t_2 à t_3 de la figure 7-41 b)].
- À partir de zéro, décompte de la fréquence inconnue pendant l'intervalle d'une seconde (t_3 à t_4); le compteur s'arrête à 237.
- Conservation et affichage du nombre 237 pendant 2 s (t_4 à t_6).

Comme le groupe des afficheurs est raccordé directement aux sorties du compteur, il reproduit la mise à zéro et le dénombrement effectué par le compteur. Il devient dans ce cas extrêmement difficile de lire la fréquence inconnue, sauf si l'intervalle d'échantillonnage est très long. Ce problème peut être éviter au moyen d'un *registre tampon* installé entre le compteur et l'unité décodeur/afficheur. Nous abordons le rôle d'un tel registre dans le problème 7-36.

QUESTIONS DE RÉVISION

1. Quel est l'intervalle d'échantillonnage le plus approprié dans le cas d'un compteur d'impulsions de quatre étages DCB et d'une fréquence à mesurer comprise entre 2 et 8 Mips?
2. Décrivez la séquence des opérations du compteur de fréquence complet de la figure 7-41.

7-17 APPLICATIONS: HORLOGE NUMÉRIQUE

Une des applications les plus courantes des compteurs est l'horloge numérique, c'est-à-dire une horloge où l'heure du jour est indiquée au moyen de chiffres: 2 chiffres pour les heures, 2 chiffres pour les minutes et 2 chiffres pour les secondes. La construction d'une horloge numérique précise exige en premier lieu une base de temps contrôlée très étroitement. Dans le cas des horloges numériques (ou des montres numériques) à pile, la base de temps est généralement fournie par un oscillateur à quartz. Les horloges numériques alimentées par le secteur utilise comme base de temps la fréquence de 60 Hz du secteur. Dans un cas comme dans l'autre, il faut que cette base de temps soit divisée pour obtenir un signal de 1 Hz, c'est-à-dire un signal d'une impulsion par seconde. La figure 7-42 illustre le schéma fonctionnel d'une horloge numérique fonctionnant d'après un signal de 60 Hz.

Le signal de 60 Hz est envoyé à un circuit conformateur pour produire des ondes carrées délivrées au rythme de 60 par seconde. Cette forme d'onde de 60 impulsions par seconde alimente un compteur MODULO-60 qui ramène ces 60 impulsions par seconde à 1 impulsion par seconde. Le signal de 1 impulsion par seconde est délivré à la section des SECONDES, qui compte et affiche les secondes de 0 à 59. Le compteur DCB est incrémenté à chaque seconde. Après 9 secondes, il est recyclé à 0 ce qui a pour effet de déclencher le compteur MODULO-6 et de l'incrémenter. Ce processus se poursuit pendant 59 secondes; à ce moment-là le compteur MODULO-6 contient le nombre 101 (5) et le compteur DCB, le nombre 1001 (9). L'affichage est donc 59 s. L'impulsion suivante recycle le compteur DCB à 0, ce qui provoque le recyclage du compteur MODULO-6 à 0 (rappelez-vous que le compteur MODULO-6 compte de 0 à 5).

La sortie du compteur MODULO-6 de la section des SECONDES a une fréquence de 1 impulsion par minute (le compteur MODULO-6 est recyclé à toutes les 60 secondes). Cette sortie est transmise à la section des MINUTES, qui compte et affiche les minutes de 0 à 59. La section des MINUTES est identique à la section des SECONDES et son fonctionnement est tout à fait analogue.

La sortie du compteur MODULO-6 des minutes a une fréquence de 1 impulsion par heure: ce compteur MODULO-6 est recyclé à toutes les 60 minutes. Cette sortie alimente la section des HEURES, qui compte et affiche les heures de 1 à 12. La section des HEURES est différente de celles des SECONDES et des MINUTES du fait qu'elle n'est jamais ramenée à l'état 00. Les circuits de cette section présentent certains aspects inhabituels justifiant une analyse plus détaillée.

La figure 7-43 nous fait voir les circuits détaillés de la section des HEURES. On y voit un compteur DCB qui compte le chiffre unité des heures, et une bascule (compteur MODULO-2) qui compte le chiffre des dizaines d'heures. Le compteur DCB est une puce 74192 qui fonctionne exactement comme la puce 74193 déjà étudiée, sauf qu'elle affiche seulement les nombres entre 0000 et 1001. Autrement dit, la puce 74192 peut compter en DCB (c'est-à-dire de 0 à 9 puis revenir à 0) ou décompter en DCB (c'est-à-dire de 9 à 0 puis revenir à 9). Dans le présent exemple, on s'en sert pour compter le signal de l'impulsion par heure qui provient de la section des MINUTES. L'INVERSEUR placé à l'entrée CP_U est nécessaire parce que la puce 74192 réagit au FM, alors que nous voulons qu'elle réagisse au FD qui se produit lorsque les compteurs de la section des MINUTES sont recyclés à 0.

Le rythme des impulsions qui arrivent au compteur DCB est de une par heure. Par exemple, à 7 heures, le chiffre dans ce compteur est 0111, et son circuit décodeur/afficheur fait voir le chiffre 7. Au même moment, la borne X est au niveau BAS de sorte que le chiffre affiché est 0. Donc l'ensemble des deux chiffres donne « 07 ». Quand le chiffre dans le compteur DCB est 1001 (9), il est recyclé au moment de l'impulsion suivante. Le

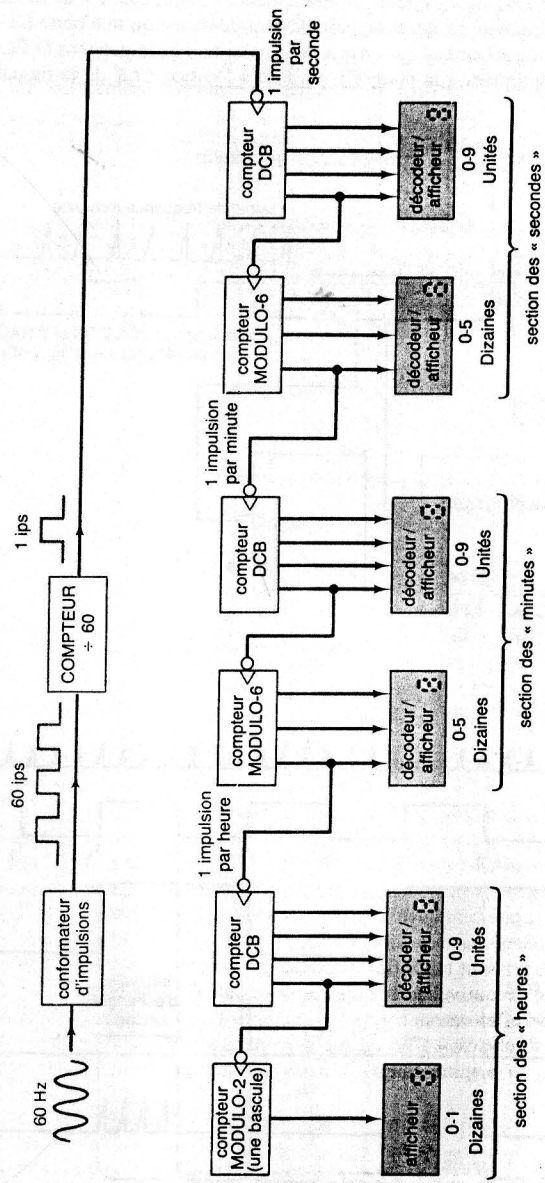


FIGURE 7-42 Schéma fonctionnel d'une horloge numérique.

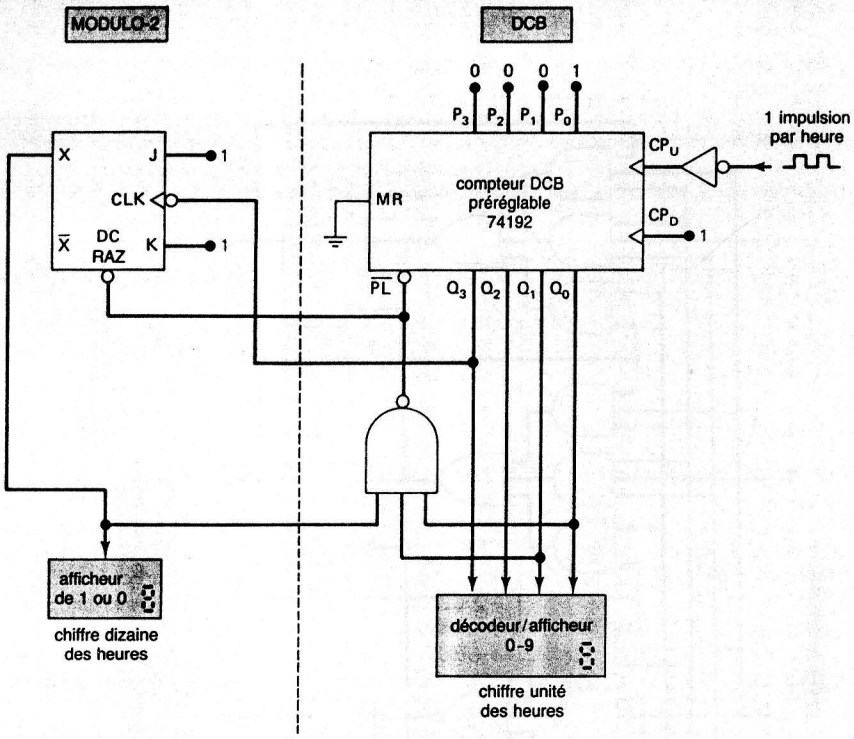


FIGURE 7-43 Circuits détaillés de la section des heures.

FD qui se produit à la borne Q_3 fait passer la bascule X de 0 à 1. L'afficheur de cette bascule présente donc maintenant le chiffre 1 et celui du compteur DCB le chiffre 0, de sorte que le nombre affiché pour les heures est « 10 », c'est-à-dire 10 heures.

Les deux impulsions suivantes qui atteignent le compteur DCB provoquent l'affichage de « 11 » et « 12 », pour indiquer respectivement 11 heures et 12 heures. L'impulsion qui suit ces deux dernières place le compteur dans l'état 0011 (3). Dans cet état, les sorties Q_1 et Q_0 du compteur sont toutes les deux à 1, tout comme X. Donc la sortie de la porte NON-ET passe à 0 ce qui valide la borne *RAZ* de la bascule X et l'entrée $\overline{PL}$ de la puce 74192. Ceci a pour effet de ramener à 0 la bascule X et de ramener le compteur DCB à 0001. Le nombre affiché est « 01 » ce qui correspond à 1 heure. On aborde dans plusieurs problèmes de la fin du chapitre d'autres aspects qui viennent compléter l'étude de l'horloge numérique.

QUESTIONS DE RÉVISION

1. Nommez les éléments constitutifs d'une horloge numérique.
2. Dites pourquoi il faut un INVERSEUR sur la figure 7-43.

7-18 REGISTRES INTÉGRÉS

Les nombreux types de registres existants peuvent être classés selon la méthode d'écriture des données dans le registre et de lecture de ces données. Voici les différents types.

1. Écriture parallèle/lecture parallèle.
2. Écriture série/lecture série.
3. Écriture parallèle/lecture série.
4. Écriture série/lecture parallèle.

Tous ces types de registres existent en version circuits intégrés; un concepteur de circuit logique peut facilement trouver ce qui convient à son application. Dans les sections suivantes, nous allons examiner un circuit représentatif de chacune de ces catégories.

7-19 ÉCRITURE PARALLÈLE/LECTURE PARALLÈLE — LES PUCES 74174 ET 74178

En réalité, il y a deux types de registres dans cette catégorie: un qui est entièrement parallèle et un autre qui est un registre à décalage pouvant être chargé avec des données parallèles et lu en parallèle.

La puce 74174 Le schéma logique de la puce 74174 (aussi 74LS174 et 74HC174) est reproduit à la figure 7-44 a): il s'agit d'un registre de 6 bits qui possède des entrées parallèles (D_5–D_0) et des sorties parallèles Q_5–Q_0. Des données parallèles sont chargées dans le registre au moment du FM du signal d'horloge *CP*. Une borne de remise à zéro principale $\overline{MR}$ sert à mettre à zéro de manière asynchrone toutes les bascules du registre.

À la figure 7-44 b) vous pouvez voir le symbole logique du 74174. Utilisé dans les schémas de circuit, ce symbole représente le registre de la figure 7-44 a).

La puce 74178 La figure 7-45 a) nous fait voir le schéma logique de la puce 74178 (aussi 74HC178), un registre à décalage de 4 bits qui possède des entrées parallèles (P_0–P_3) et des sorties parallèles (Q_0–Q_3). Toutefois il possède une borne de donnée série, D_S, et deux bornes de validation: *PE* (validation parallèle) et *SE* (validation série).

À la figure 7-45 b), on a reproduit la table de sélection des modes qui nous montre comment choisir les divers modes d'opération de ce CI. La première ligne nous donne les conditions pour réaliser le décalage vers la droite. Quand $SE = 1$, les données sont décalées de gauche à droite au moment du FD de l'impulsion d'horloge $\overline{CP}$, et cela quel que soit le niveau logique sur l'entrée *PE* (rappelons qu'un *X* signifie une condition « indifférente »). On peut vérifier l'exactitude de ceci en suivant le cheminement des signaux dans le schéma logique et en notant que lorsque $SE = 1$, le signal sur D_S traverse les portes logiques et se retrouve à l'entrée D de la bascule Q_0. De même, le signal de Q_0 se retrouvera sur l'entrée D de Q_1, le signal de Q_1 sur l'entrée D de Q_2; et le signal de Q_2 sur l'entrée D de Q_3.

La seconde ligne de cette table donne les conditions qui autorisent un transfert parallèle des données placées sur les entrées parallèles (P_0–P_3) vers les sorties parallèles (Q_0–Q_3). Quand $SE = 0$ et $PE = 1$, le transfert parallèle est effectué au moment du FD de $\overline{CP}$. Notez qu'il s'agit ici d'un transfert *synchrone*. De nouveau, en suivant le cheminement des signaux dans le schéma logique, vous constatez que lorsque $SE = 0$ et $PE = 1$, les signaux sur les entrées parallèles traversent les portes logiques et se retrouvent sur les entrées D de leur bascule respective.

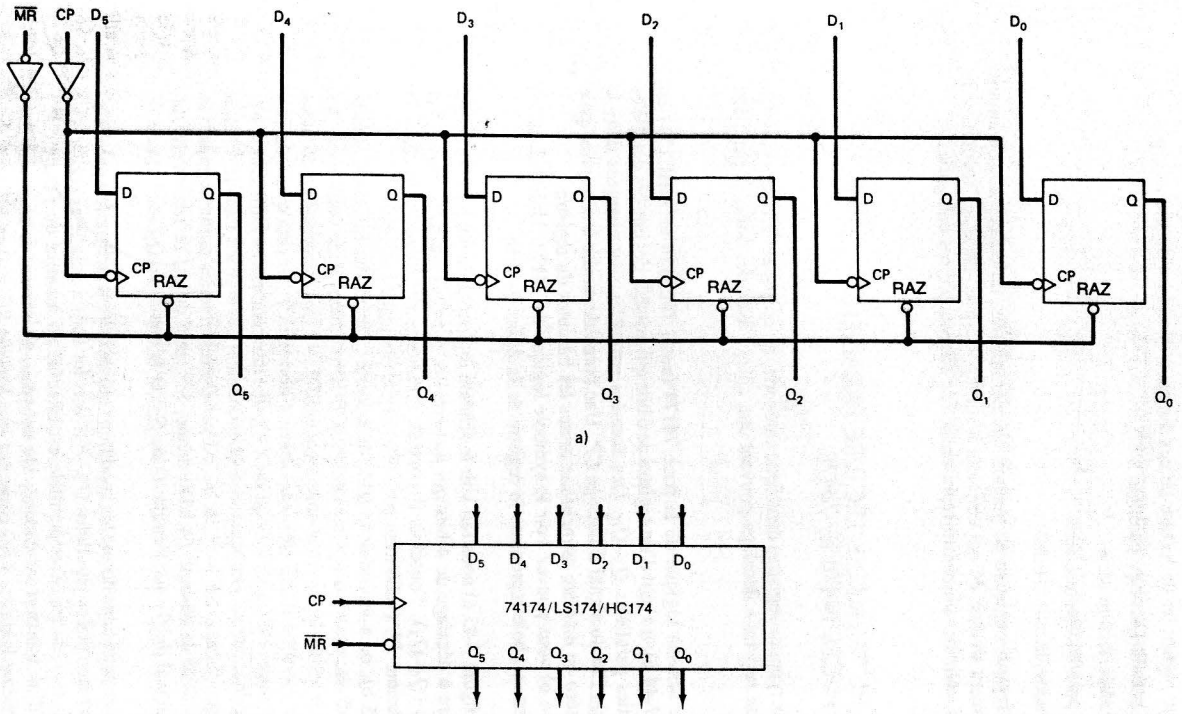


FIGURE 7-44 a) Schéma de la puce 74174; b) symbole logique. (Gracieuseté de Fairchild, une filiale Schlumberger).

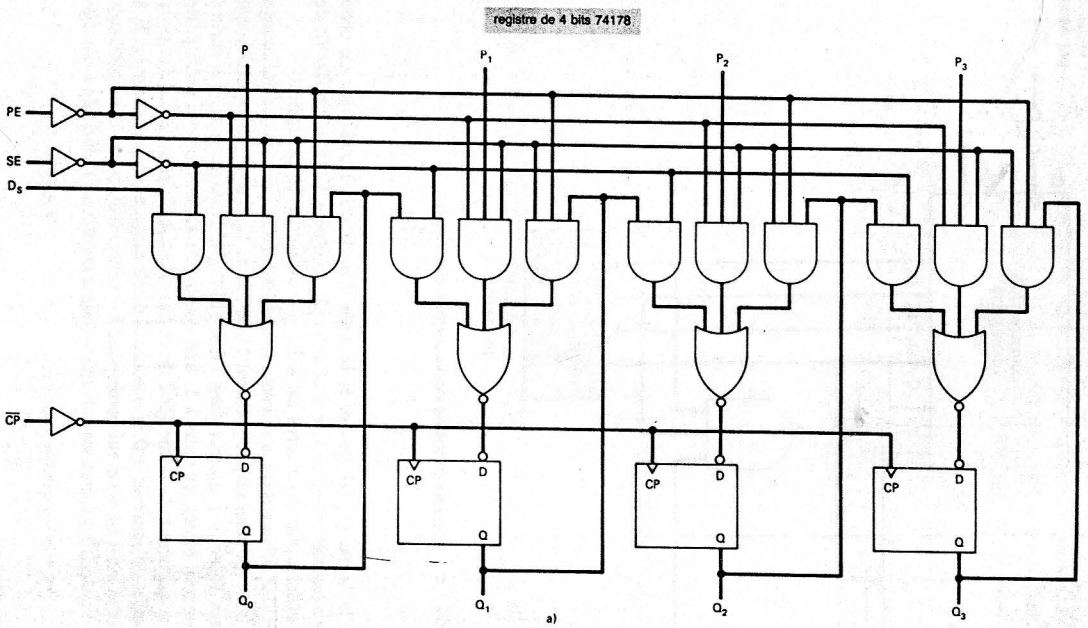


Table de sélection des modes

Entrées			Réponse
SE	PE	$\overline{CP}$	
H	X		décalage vers la droite $D_S \rightarrow Q_0$; $Q_0 \rightarrow Q_1$, etc.
L	H		chargement parallèle $P_0 \rightarrow Q_0$; $P_1 \rightarrow Q_1$, etc.
L	L	X	maintien

H = niveau de tension HAUT
L = niveau de tension BAS
X = indifférent

b)

FIGURE 7-45 a) Schéma logique du 74178; b) table de sélection des modes. (Gracieuseté de Fairchild, une filiale Schlumberger).

La dernière ligne de cette table nous apprend que lorsque $SE = 0$ et $PE = 0$, les bascules du registre conservent les valeurs courantes, quelle que soit la condition qui existe sur l'entrée du signal d'horloge. Dans cette condition d'entrée, la sortie de chaque bascule passe à travers les portes logiques qui sont associées à cette bascule et se retrouve sur l'entrée D de la même bascule. Donc l'arrivée d'un FD à la borne $\overline{CP}$ ne change pas l'état de la bascule.

La figure 7-46 nous montre le symbole logique de la puce 74178/74HC178.

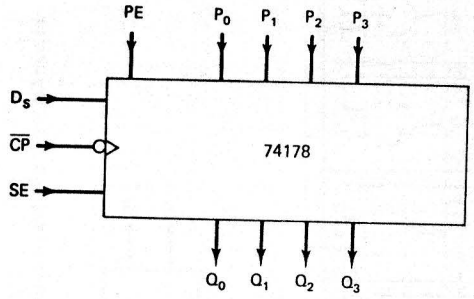


FIGURE 7-46 Symbole logique de la puce 74178.

EXEMPLE 7-20

Décrivez la façon de raccorder la puce 74178 pour obtenir un compteur en anneau.

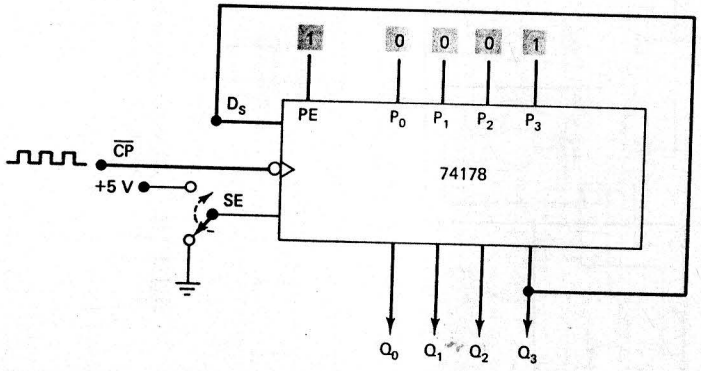


FIGURE 7-47 Puce 74178 câblée pour réaliser un compteur en anneau.

Solution Les connexions nécessaires sont illustrées à la figure 7-47. La sortie Q_3 est ramenée à l'entrée D_S. Les entrées parallèles sont fixées en permanence à 0001, et l'entrée *PE* est continuellement au niveau HAUT. L'entrée *SE* est mise au niveau BAS au moment initial. Ainsi, à chaque FD de $\overline{CP}$, le registre à décalage est chargé en parallèle par 0001.

Ensuite, *SE* est mis au niveau HAUT (mode décalage) et reste dans cet état pendant que les fronts descendants du signal d'horloge font circuler le 1 dans le registre.

7-20 ÉCRITURE SÉRIE/LECTURE SÉRIE — LA PUCE 4731B

La puce 4731B est un circuit intégré de la technologie CMOS, et elle comprend *quatre* registres à décalage de 64 bits. La figure 7-48 reproduit le schéma logique d'un des registres de 64 bits. Il possède une entrée série, D_S, une borne d'horloge $\overline{CP}$ qui réagit au FD et une sortie série dans la dernière bascule, Q_{63}. C'est la seule sortie accessible de l'extérieur. Notez que cette sortie est raccordée à un circuit *tampon* (le symbole triangle sans le petit rond d'inversion). Un tampon ne change pas le niveau logique du signal; on s'en sert pour obtenir une capacité de courant de sortie plus élevée que normale. En outre remarquez qu'il n'y a aucune façon d'écrire en parallèle des données dans les bascules des registres.

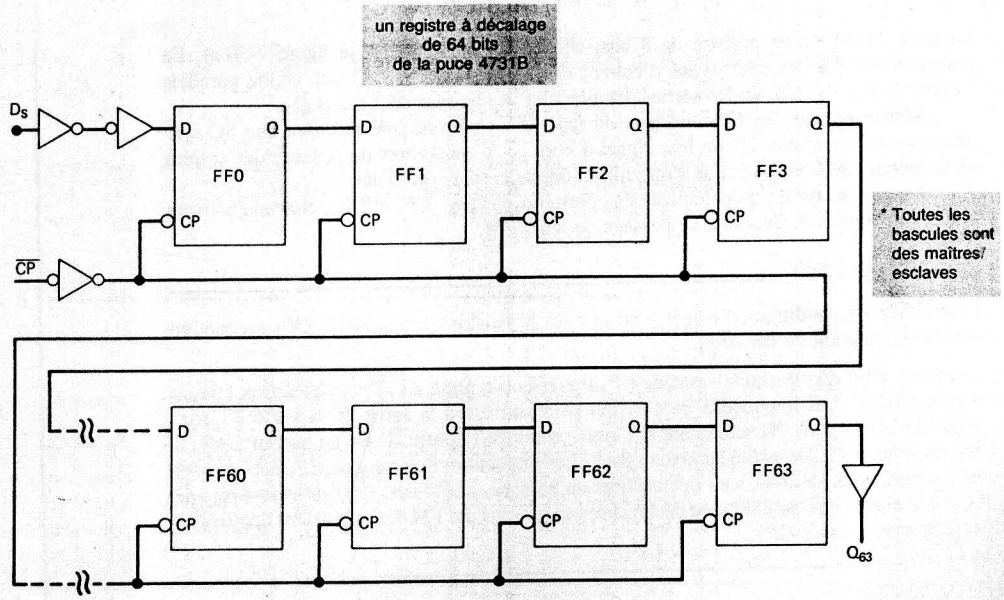


FIGURE 7-48 Schéma logique d'un des registres à décalage de 64 bits qu'on trouve sur la puce 7431B. (Gracieuseté de Fairchild, une filiale Schlumberger).

EXEMPLE 7-21

Dites comment on peut câbler la puce 4731B pour réaliser un registre à décalage de 256 bits.

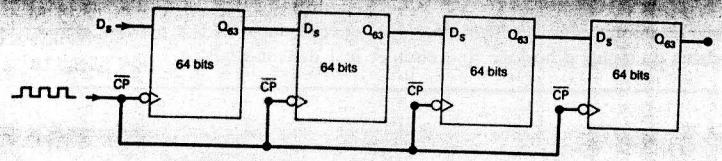


FIGURE 7-49 On peut connecter ensemble les quatre registres à décalage de 64 bits de la puce 4731B pour obtenir un registre de 256 bits.

Solution Il est possible de réunir ensemble les quatre registres à décalage pour former un seul registre, en raccordant la sortie Q_{63} d'un registre à l'entrée D_S du registre suivant et en raccordant ensemble toutes les entrées $\overline{CP}$ afin d'obtenir un signal d'horloge commun. C'est ce que nous montre la figure 7-49; sur celle-ci on a utilisé le symbole logique du registre à décalage pour plus de commodité.

7-21 ÉCRITURE PARALLÈLE/LECTURE SÉRIE — LA PUCE 74165/74LS165/74HC165

La puce 74165 est un registre de 8 bits; ce registre est illustré à la figure 7-50 a). En réalité, il possède les deux types d'entrée: une entrée série, D_S, et une entrée parallèle formée de P_0–P_7. Les seules sorties des bascules accessibles sont Q_7 et $\overline{Q}_7$.

Remarquez que les bascules sont du type SR synchrone (même chose que SC synchrone), réagissant aux FD de leur signal d'horloge. Chaque bascule possède des entrées asynchrones, *RAU* et *RAZ*, que l'on utilise comme entrées parallèles.

Remarquez qu'il y a également deux bornes d'horloge, CP_1 et CP_2, chacune pouvant lancer l'opération de décalage au passage de leurs FM.

EXEMPLE 7-22

Déterminez les conditions d'entrée pour que les impulsions appliquées à CP_1 mettent en marche l'opération de décalage.

Solution Pour que les impulsions de CP_1 traversent la porte ET, l'entrée $\overline{PL}$ doit être au niveau HAUT. Les impulsions de CP_1 qui se retrouvent à la sortie de la porte ET sont inversées par la porte NI seulement si l'autre entrée de la porte NI est au niveau BAS. Il faut donc que CP_2 soit gardé au niveau BAS. Les impulsions inversées de CP_1 déclenchent les bascules. Les bascules sont déclenchées au passage du FD des impulsions inverses de CP_1, ce qui veut dire qu'elles sont déclenchées au passage des FM des impulsions appliquées sur la borne d'entrée CP_1.

EXEMPLE 7-23

Dites comment il est possible de mettre la puce 74165 en mode « maintien » pour que le registre conserve le même contenu quand arrivent des impulsions sur la borne CP_1.

Solution Si CP_2 est gardé au niveau HAUT pendant que $\overline{PL}$ est au niveau HAUT, la sortie de la porte ET inférieure reste au niveau HAUT et maintient continuellement la sortie de la porte NI à 0, quelle que soit la valeur de CP_1.

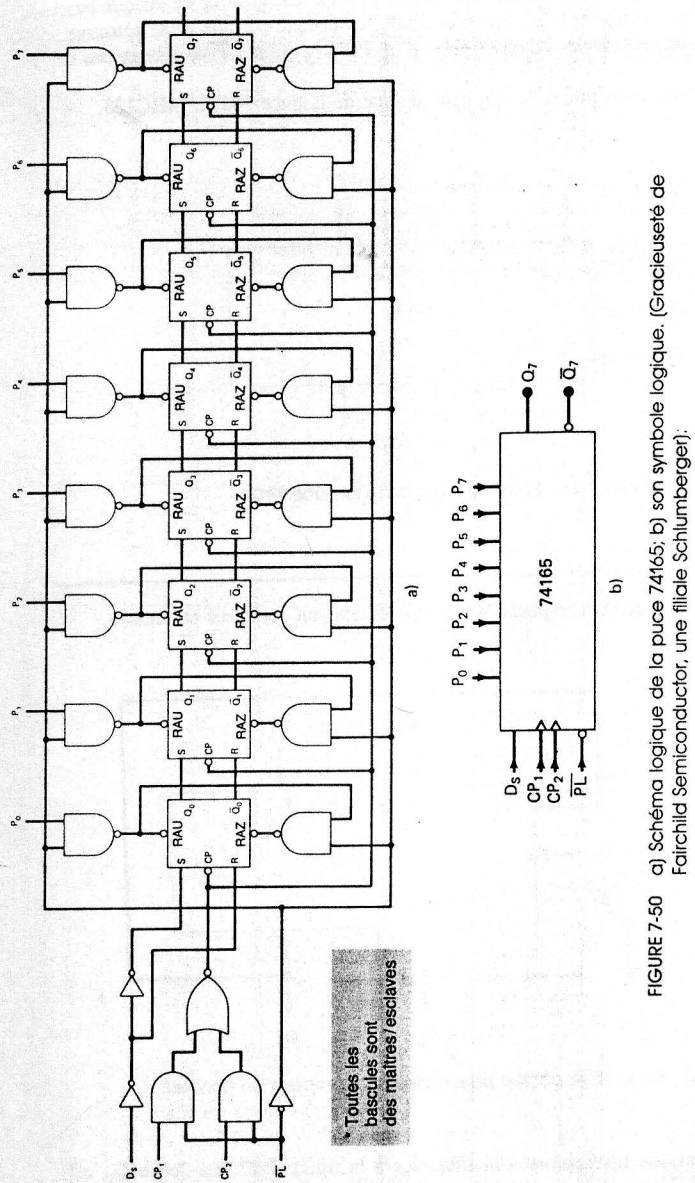


FIGURE 7-50 a) Schéma logique de la puce 74165; b) son symbole logique. (Gracieuseté de Fairchild Semiconductor, une filiale Schlumberger).

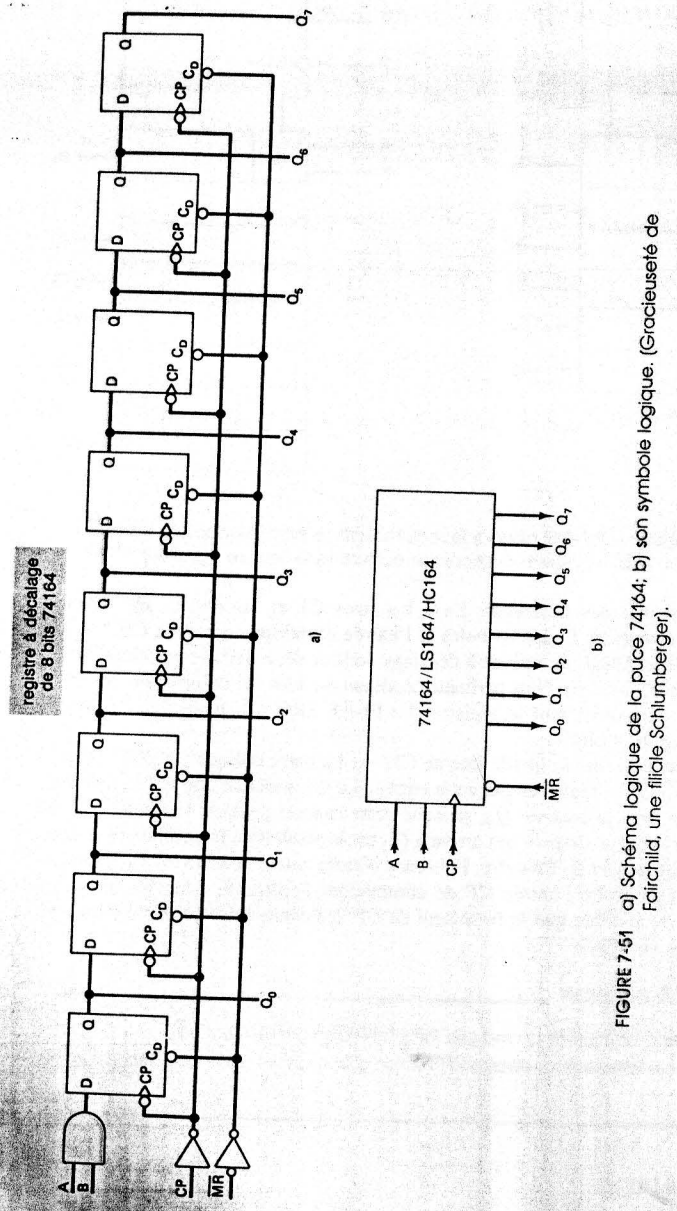


FIGURE 7-51 a) Schéma logique de la puce 74164; b) son symbole logique. (Gracieuseté de Fairchild, une filiale Schlumberger).

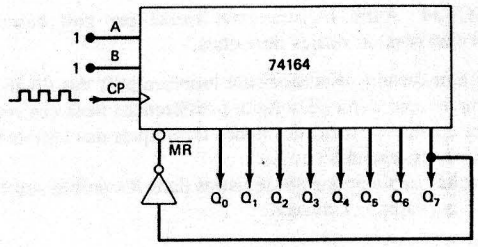
7-22 ÉCRITURE SÉRIE/LECTURE PARALLÈLE — LA PUCE 74164/74LS164/74HC164

Le schéma logique de la puce 74164 est reproduit à la figure 7-51 a). Il s'agit d'un registre à décalage de 8 bits dont chacune des bascules est accessible de l'extérieur. Dans cette puce, plutôt que d'avoir une borne d'entrée série, on combine dans une porte ET les entrées A et B pour produire une borne série sur la bascule Q_0.

L'opération de décalage se produit au passage des FM du signal d'horloge CP. La borne $\overline{MR}$ permet de remettre à 0 de manière asynchrone toutes les bascules lorsqu'on lui applique un niveau BAS.

EXEMPLE 7-24

Supposez que le contenu initial du registre 74164 de la figure 7-52 a) soit 00000000. Déterminez la suite des états correspondant à la séquence des impulsions d'horloge.



a)

Impulsion d'entrée	Q_0	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7
0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0
3	1	1	1	0	0	0	0	0
4	1	1	1	1	0	0	0	0
5	1	1	1	1	1	0	0	0
6	1	1	1	1	1	1	0	0
7	1	1	1	1	1	1	1	0
8	1	1	1	1	1	1	1	1

recyclage

état transitoire

b)

FIGURE 7-52 Exemple 7-24.

Solution La suite recherchée est donnée à la figure 7-52 b). Quand $A = B = 1$, l'entrée série est à 1, de sorte qu'un chiffre 1 est décalé dans le registre à chaque FM de CP. Étant donné que Q_7 est au début à 0, l'entrée $\overline{MR}$ n'a pas d'effet.

À l'arrivée de la huitième impulsion, le registre tente de passer à 11111111 lorsque le 1 se trouvant dans Q_6 est décalé dans Q_7. Cet état ne se produit que momentanément, car au moment où Q_7 passe à 1, un signal BAS sur $\overline{MR}$ ramène immédiatement le registre

à l'état initial 00000000. La suite des états indiqués recommence pour les huit impulsions suivantes.

Voici une liste de certains autres registres intégrés qui sont des variantes de ceux déjà décrits:

74194/LS194/HC194 *Registre intégré à décalage universel bidirectionnel de 4 bits* qui décale à gauche, à droite, accepte des données parallèles et fournit des données parallèles. Ces opérations sont sélectionnées par un code de sélection de mode de 2 bits, délivré sur des entrées du compteur. Le problème 7-47 vous donne l'occasion d'en apprendre davantage sur ce circuit.

74373/LS373/HC373/HCT373 Voici un registre à 8 bits (octal) contenant huit éléments de mémoire *D* dotés de sorties *trois états*. Une sortie trois états est un type spécial de sortie de circuit logique qui permet la réunion de plusieurs sorties de ce genre. Nous introduisons les sorties trois états dans le chapitre suivant.

74374/LS374/HC374/HCT34 Autre registre octal formé par huit bascules *D* déclenchées par un signal et dotées de sorties trois états.

Les registres intégrés que nous venons d'étudier sont représentatifs des divers types que l'on retrouve dans le commerce. En dépit de certaines différences dans ces registres de base, vous êtes maintenant en mesure, à partir des fiches techniques des fabricants, de comprendre comment fonctionnent les registres.

Vous trouverez plusieurs autres applications des registres dans les problèmes à la fin du chapitre, de même que dans les chapitres suivants.

QUESTIONS DE RÉVISION

1. Quel genre de registre peut accueillir en une seule fois la totalité d'un nombre binaire et le restituer à l'extérieur un bit à la fois?
2. *Vrai ou faux:* Un registre écriture série, lecture parallèle peut voir afficher ses bits en une seule fois.
3. Quel genre de registre reçoit les bits un à la fois, mais les restitue sur ses sorties en une seule fois?
4. Dans quel type de registre n'avons-nous accès qu'aux seuls bits de gauche et de droite?

7-23 SYMBOLES IEEE/ANSI POUR LES REGISTRES*

Nous nous proposons de traiter deux exemples de symboles IEEE/ANSI pour les registres intégrés. En premier lieu, considérons le CI 74174, un registre à écriture parallèle/lecture parallèle, dont la logique interne et le symbole logique traditionnel sont illustrés à la figure 7-44. Le symbole logique IEEE/ANSI du 74174 est reproduit à la figure 7-53 a). Son périmètre englobe le bloc de commande commun avec les deux renfoncements et six rectangles étroits représentant les six bascules.

Le bloc de commande commun regroupe les entrées communes à tous les éléments de la puce: dans ce cas-ci, les entrées $\overline{MR}$ et *CP* qui alimentent les six bascules Q_0 à Q_5 constituant le registre. Le libellé interne associé à l'entrée $\overline{MR}$ est un R qui indique que la fonction de cette borne est la *remise à zéro* de chaque bascule. Le libellé interne associé à la borne *CP* est C1, ce qui nous informe que cette entrée contrôle l'entrée de données dans tout élément de mémoire ayant le préfixe 1 dans son libellé. Chaque entrée *D* des bascules a comme libellé interne 1D (indiqué seulement pour la bascule Q_0, mais qui

* Cette section peut être omise sans rien enlever à la continuité de la présentation.

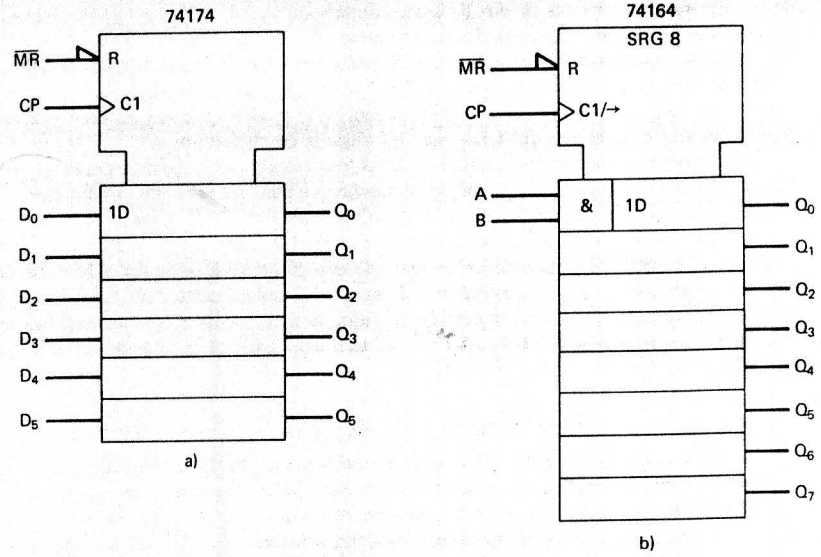


FIGURE 7-53 Symboles IEEE/ANSI pour a) le registre écriture parallèle/lecture parallèle 74174, et b) le registre à décalage écriture série/lecture parallèle 74164.

s'entend pour l'ensemble des bascules). Le « 1 » dans C1 et 1D établit une relation fonctionnelle entre les entrées *D* des bascules et l'entrée d'horloge commune *CP*.

Le symbole IEEE/ANSI du registre à décalage écriture série/lecture parallèle 74164 est reproduit à la figure 7-53 b). Son périmètre contient un bloc de commande commun et les huit bascules qui constituent le registre. Le libellé SRG 8 signale que ce boîtier intégré est un registre de 8 bits.

L'entrée *CP* est associée au libellé interne C1/→. La barre oblique (/) sert à séparer les deux fonctions C1 et → réalisées par cette entrée. Le C1 annonce que *CP* commande l'entrée des données dans la bascule Q_0, puisque cette bascule possède le libellé interne 1D. Remarquez que le bit de donnée qui arrive à Q_0 est le résultat de la combinaison dans une porte ET des entrées A et B. En outre, l'absence d'autre entrée externe sur les bascules restantes (Q_1 à Q_7) empêche l'entrée *CP* de commander l'entrée de données dans ces dernières. La flèche → indique que le front actif de *CP* déclenche l'opération de décalage vers la droite (de Q_0 vers Q_7).

QUESTIONS DE RÉVISION

1. Que signifie la barre oblique (/) quand elle fait partie d'un libellé d'entrée?
2. Quelle notation sera utilisée pour décrire la fonction d'un registre de la puce 4731B, illustrée à la figure 7-48.

7-24 DÉPANNAGE

Les bascules, les compteurs et les registres sont les principaux composants des **systèmes logiques séquentiels**. Un système logique séquentiel, du fait de ses éléments de mémoire,

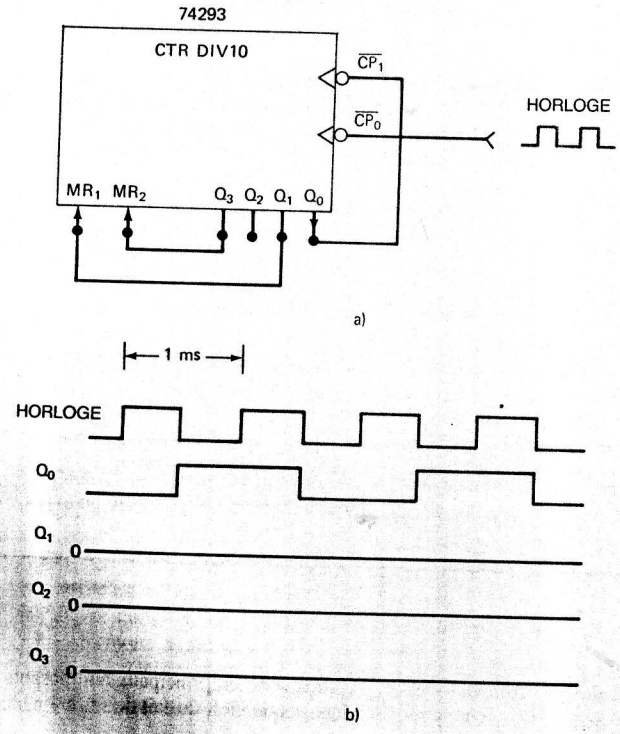
a de façon caractéristique des valeurs de sortie et une séquence d'opérations qui dépendent des valeurs actuellement sur les entrées et des entrées survenues précédemment. Même si, généralement, les circuits séquentiels sont plus complexes que les circuits combinatoires, les procédures de dépannage déductives s'appliquent aussi bien à ces deux genres de circuits. Les systèmes séquentiels subissent les mêmes genres de pannes (circuits ouverts, courts-circuits, défaillances internes des CI...) que les circuits combinatoires.

Le travail de dépistage des défaillances dans les systèmes séquentiels emprunte, tout au moins dans ses grandes lignes, la même démarche que dans les systèmes combinatoires. Une des techniques de dépannage les plus efficaces a pour point de départ l'observation par le technicien du fonctionnement du circuit, puis, par raisonnement, la détermination des causes possibles pouvant entraîner ce genre de dérèglement. Ensuite, le technicien ou la technicienne essaie avec des instruments de mesure d'isoler la défaillance exacte. Les exemples qui suivent montrent comment on applique la méthode du raisonnement dans le dépannage de systèmes séquentiels. Après l'étude de ces exemples, vous aurez tous les outils pour résoudre les problèmes de dépannage de la fin de ce chapitre.

EXEMPLE 7-25

La figure 7-54 a) illustre un 74293 câblé comme un compteur MODULO-10. Un technicien teste ce compteur en appliquant un signal d'horloge de 1 Hz et en observant les sorties Q à l'oscilloscope. Les formes d'ondes observées sont montrées sur la figure 7-54 b). Déterminez les causes possibles de ce fonctionnement défectueux.

FIGURE 7-54 Exemple 7-25.



Solution Les formes d'ondes montrent que Q_0 bascule en réponse au front descendant de l'horloge, mais que toutes les autres bascules restent en permanence à l'état BAS. Il y a différentes causes possibles à un tel dérèglement.

1. La sortie Q_1 est court-circuitée à la masse par l'intérieur ou l'extérieur. Reportez-vous à la figure 7-8 a) et constatez qu'une telle éventualité a pour effet d'empêcher Q_2 et Q_3 de basculer, étant donné que Q_1 est le signal d'horloge de Q_2 et que Q_2 est l'horloge de Q_3.
2. La connexion entre Q_0 et $\overline{CP}_1$ est ouverte, ce qui empêche Q_1 de recevoir le signal d'horloge.
3. Il y a une panne à l'intérieur de cette puce qui freine le basculement de Q_1.

EXEMPLE 7-26

Après l'analyse de la situation illustrée au cours de l'exemple 7-25, le technicien s'efforce d'isoler la défaillance. Il effectue des mesures à l'ohmmètre et vérifie que Q_1 n'est pas court-circuité à la masse et que Q_0 est bien raccordé à $\overline{CP}_1$. Ces constatations éliminent les deux premières défaillances envisageables. Il lui reste à conclure que la puce est défectueuse et à remplacer cette dernière. À sa grande surprise, ce circuit continue de manifester les mêmes symptômes. Embarrassé par tout ceci, il décide d'examiner les formes d'ondes des bascules plus attentivement en utilisant l'échelle de temps 10 ns/cm. À cette échelle, il remarque un léger décrochement dans le signal Q_1 à l'instant où Q_0 passe par un front descendant (voyez figure 7-55). Quelle est la défaillance probable?

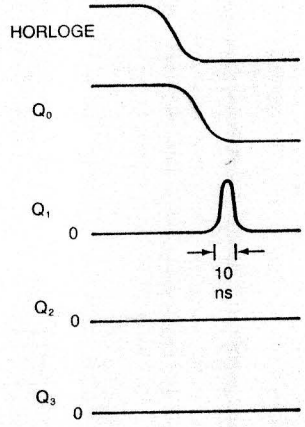


FIGURE 7-55 Exemple 7-26.

Solution Quand le compteur fonctionne sans problème, on doit observer un décrochement en Q_1 quand s'effectue la transition vers 1010 (dix), lorsque les niveaux HAUT sur Q_3 et Q_1 commandent aux entrées MR de remettre le compteur à 0000. Les formes d'ondes reproduites à la figure 7-55 ne font pas état, toutefois, d'une valeur HAUTE sur Q_3 au moment de l'arrivée du décrochement de Q_1. La défaillance la plus probable est une connexion ouverte en MR_2, puisque dans un tel cas, pour un circuit intégré TTL, cela est équivalent à une valeur HAUTE constante. Par conséquent, dès que Q_1 passe à 1, les entrées MR sont toutes les deux au niveau HAUT, et le compteur est remis à l'état 0000.

EXEMPLE 7-27

Une technicienne soumet à un test le compteur de fréquence de la figure 7-41 et cela pour différentes valeurs de l'intervalle d'échantillonnage et diverses fréquences d'entrée inconnues. Dans tous les cas, elle relève que la fréquence affichée est toujours exactement *deux* fois ce qu'elle devrait être. Quelle est la cause possible de ce dérèglement?

Solution Reportez-vous à la figure 7-41 et constatez que la fréquence inconnue est autorisée à franchir la porte ET et à arriver dans le compteur durant l'intervalle entre t_3 et t_4, quand les bornes ÉCHANTILLONNAGE et X sont toutes les deux à 1. Si l'entrée médiane de la porte ET est ouverte, elle se comportera alors comme un niveau HAUT permament (cas d'un élément TTL). Le déblocage de la porte ET pour laisser passer les impulsions de fréquence ne se produit que lorsque X est au niveau HAUT durant l'intervalle t_2 à t_4. C'est un intervalle deux fois plus long que l'intervalle normal, et le compteur dénombre donc deux fois plus d'impulsions.

EXEMPLE 7-28

Un technicien établit le câblage de l'horloge numérique des figures 7-42 et 7-43. Il observe que la section des secondes compte parfaitement bien. Pour tester rapidement le fonctionnement des sections des minutes et des heures, il contourne le compteur MODULO-60 afin que les compteurs soient déclenchés à un rythme 60 fois plus rapide que normal. Ses observations concluent à un bon fonctionnement de la section des minutes, mais montrent que la section des heures dénombrent comme ceci:

Dizaines des heures	Unités des heures
0	1
0	2
0	3
0	4
0	5
0	6
0	7
1	8
1	9
1	0
1	1
1	2

Quelle est la cause probable de cette défectuosité?

Solution Comme la difficulté se situe dans la section des heures, reportons-nous à la figure 7-43. La séquence illustrée ci-dessus est exacte sauf que le chiffre des dizaines passe de 0 à 1 quand le chiffre des unités passe de 7 à 8, au lieu de le faire quand ce dernier passe de 9 à 0. Ce genre de situation devrait se produire si l'entrée *CLK* de la bascule *X* était connectée à Q_2 plutôt qu'à Q_3 du compteur DCB. Si c'est bien ce le cas, quand le compteur DCB passe de 7 à 8, la bascule Q_2 passe de 1 à 0 ce qui fait commuter la bascule *X* avant son temps.

PROBLÈMES

SECTION 7-1

7-1 Une onde carrée de 8 MHz synchronise un compteur à propagation de 5 bits. Dites quelle est la fréquence du signal apparaissant à la sortie de la dernière bascule. Dites quel est le facteur d'utilisation de ce signal.

7-2 Même problème que 7-1, mais cette fois-ci avec un signal d'entrée ayant un facteur d'utilisation de 20 %.

7-3 Supposez qu'un compteur binaire de 5 bits a comme état initial 00000. Dites quel sera son contenu après 144 impulsions d'entrée.

SECTION 7-2

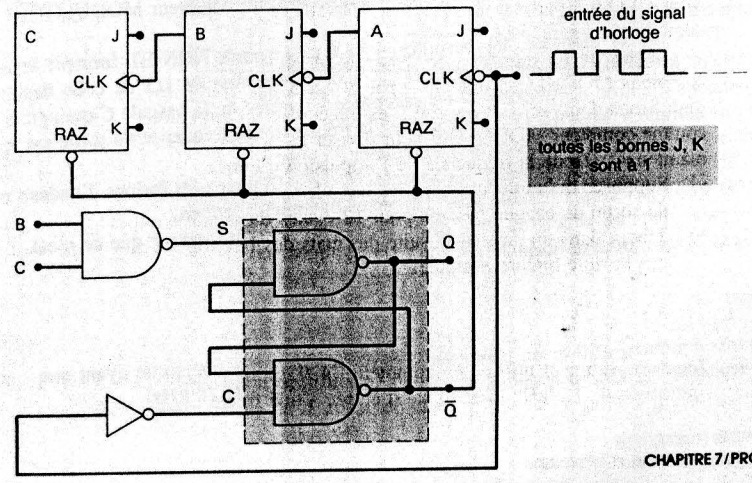
7-4 Utilisez des bascules J-K et d'autres circuits logiques pour construire un compteur asynchrone MODULO-24.

7-5 Tracez les formes d'ondes de toutes les bascules du compteur à décade de la figure 7-6 b) qui correspondent à un signal d'entrée d'une fréquence de 1 kHz. Montrez tous les signaux transitoires qui peuvent survenir sur les sorties des bascules. Déterminez la fréquence de la sortie *D*.

7-6 Quand on utilise une porte NON-ET pour ramener un compteur à 0, comme aux figures 7-4 à 7-7, il se peut que des problèmes surviennent en raison des retards de propagation entre l'entrée *RAZ* et la sortie de la bascule qui ne sont pas les mêmes d'une bascule à l'autre. Par exemple, supposons qu'une bascule est remise à zéro en 10 ns et qu'une autre l'est en 50 ns. Quand la bascule la plus rapide atteint 0, elle place la sortie de la porte NON-ET à 1, ce qui bloque l'effet de l'impulsion *RAZ* et empêche la bascule la plus lente d'être remise à 0. C'est un problème particulièrement aigü lorsque les sorties des bascules ne sont pas toutes chargées de la même façon, car, en général, le retard de propagation d'une bascule augmente avec la charge qu'elle alimente.

Une des façons d'éliminer ce problème est illustrée à la figure 7-56 dans le cas d'un compteur MODULO-6. La bascule S-C est la bascule en portes NON-ET

FIGURE 7-56



rétrocouplées déjà étudiée au chapitre 5. Analysez le fonctionnement de ce circuit et montrez comment la remise à 0 est effectuée sans que survienne le problème mentionné ci-dessus.

7-7 Reportez-vous au compteur illustré à la figure 7-57. Dites pourquoi c'est un décompteur. Ce compteur a été modifié de façon à ne *pas* compter toute la suite des nombres de 111 à 000. Trouvez la suite des états qu'il compte.

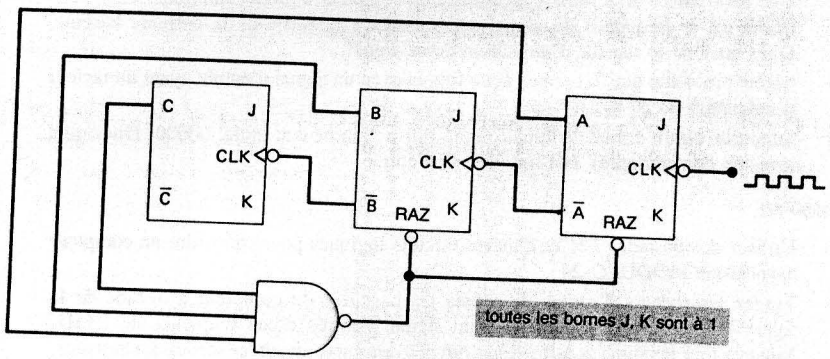


FIGURE 7-57

7-8 On utilise souvent un compteur ou un groupe de compteurs pour diviser un signal d'horloge de haute fréquence et obtenir un signal de basse fréquence. Lorsque les compteurs utilisés sont binaires (c'est-à-dire qu'ils dénombrent la suite des nombres binaires), l'onde de sortie ne sera pas une onde carrée symétrique si on a modifié la suite comptée afin d'obtenir un MODULO particulier. Par exemple, reportez-vous à la forme d'onde *C* du compteur MODULO-6 de la figure 7-4

Quand on utilise un compteur essentiellement pour la division de fréquence, il n'est pas nécessaire qu'il dénombre la suite binaire tant que nous avons le MODULO recherché. On peut obtenir une onde carrée symétrique quand le MODULO est *pair* en décomposant le MODULO en un produit de deux MODULO dont l'un est une puissance de 2. Par exemple, on peut former un compteur MODULO-6 à partir d'un compteur MODULO-3 et d'un compteur MODULO-2, comme à la figure 7-58.

Dans un tel montage, les bascules *A* et *B* et la porte NON-ET forment le compteur MODULO-3 dont la sortie *B* est à une fréquence de 1/3 de celle des impulsions d'entrée. Cette sortie *B* est raccordée à l'entrée de la bascule *C* qui agit comme compteur MODULO-2 pour diviser la fréquence; la fréquence de sortie est maintenant 1/6 de la fréquence de l'impulsion d'entrée.

a) Supposez qu'au début toutes les bascules soient à 0; tracez les formes d'ondes de sortie de chaque bascule pour 12 cycles du signal d'entrée.

b) Tracez dans un tableau la suite des états des bascules et montrez que ce n'est pas la suite binaire normale.

SECTION 7-3

7-9 Montrez comment utiliser un compteur 74293 pour produire une sortie ayant une fréquence de 1,2 kHz si le signal d'entrée a une fréquence de 18 kHz.

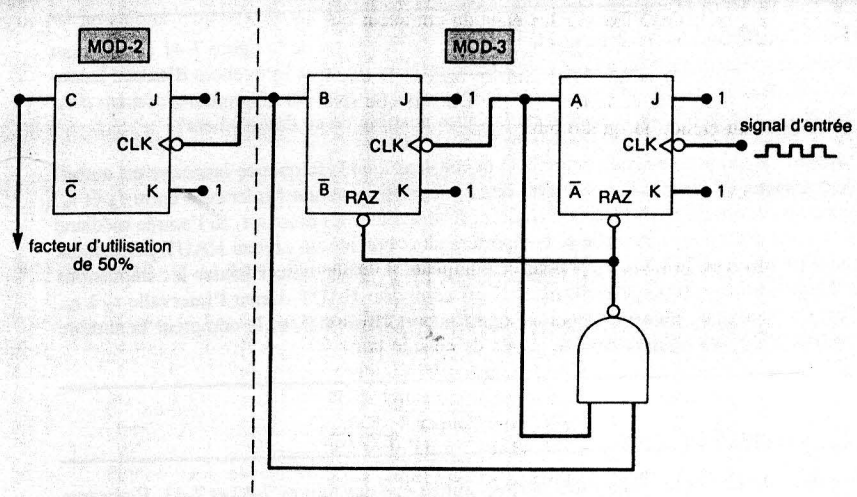


FIGURE 7-58

7-10 Montrez comment connecter deux puces 74293 pour diviser une fréquence d'entrée par 60 tout en ayant en sortie une onde carrée symétrique.

7-11 Déterminer la fréquence du signal de sortie en *X* de la figure 7-59.

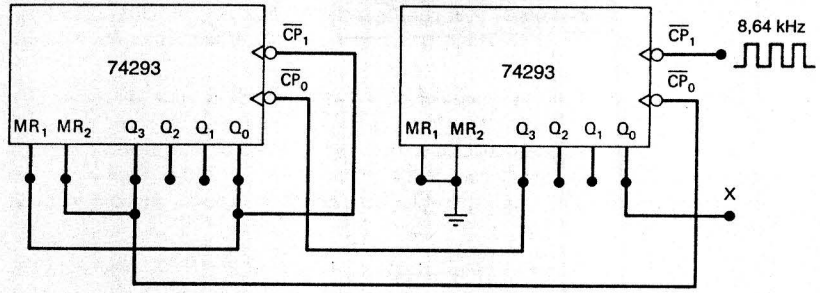


FIGURE 7-59

SECTION 7-4

7-12 a) Dessinez le schéma d'un *décompteur* MODULO-16.

b) Construisez le graphe des états.

c) Si le nombre initial du compteur est l'état 0110, quel est l'état du compteur après 37 impulsions?

SECTION 7-5

7-13 Un compteur à propagation de 4 bits est piloté par un signal d'horloge de 20 MHz. Tracez les formes d'ondes de la sortie de chaque bascule si, pour chaque bascule

t_{pd} = 20 ns. Trouvez les états du compteur qui ne surviendront jamais en raison des retards de propagation.

7-14 Indiquez la fréquence d'horloge maximale que l'on peut utiliser avec le compteur du problème de la figure 7-13. Dites quelle serait la nouvelle f_{max} si le compteur était étendu jusqu'à 6 bits.

SECTION 7-6

7-15 a) Tracez le schéma du circuit d'un compteur parallèle MODULO-64.

b) Déterminez f_{max} pour ce compteur si chaque bascule a t_{pd} = 20 ns et chaque porte a t_{pd} = 10 ns.

c) Un montage de portes logiques plus efficace peut être utilisé pour ce compteur en prenant comme entrée de chaque porte ET la sortie de la porte précédente (voir figure 7-60). Par exemple, la sortie de la première porte ET (égale à $A \cdot B$) peut être appliquée à une entrée de la deuxième porte ET, avec le signal C, sur l'autre entrée, pour donner $A \cdot B \cdot C$. De même, la sortie de la deuxième porte ET (égale à $A \cdot B \cdot C$) est délivrée à une entrée de la troisième porte ET, tandis que le signal D est appliqué à l'autre entrée pour donner en sortie $A \cdot B \cdot C \cdot D$, et ainsi de suite. Un tel montage nécessite seulement des portes ET à deux entrées. Cependant, le retard de propagation total est plus grand, puisque les signaux de sortie des bascules doivent maintenant traverser chacune des portes. Calculez le retard total pour ce montage, puis calculez f_{max} et comparez cette valeur à la valeur trouvée en b). Supposez que s'appliquent les mêmes retards que ceux donnés en b).

7-16 La figure 7-61 reproduit un compteur parallèle de 4 bits conçu de façon à ne pas passer toute la suite des 16 états binaires possibles. Analysez son fonctionnement en traçant les formes d'ondes pour chaque sortie des bascules. Ensuite, déterminez la suite des nombres de ce compteur. Supposez qu'au départ toutes les bascules sont à 0.

7-17 Montrez comment on peut connecter deux 74193 afin de réaliser un compteur à 8 bits qui divise la fréquence d'horloge par 100. Servez-vous de la sortie $\overline{TC}_D$ d'un des étages 74193 comme horloge du second étage. Raccordez les entrées de données parallèles afin d'initialiser ce compteur combiné à la valeur 100_{10}. Remarquez que ce compteur doit être préréglé quand les deux étages atteignent simultanément leur nombre final.

SECTIONS 7-8 et 7-9

7-18 La figure 7-62 montre comment un décompteur préréglable peut être utilisé comme *minuterie programmable*. La fréquence du signal d'horloge est précisément 1 Hz; cette fréquence est dérivée de la fréquence de 60 Hz du secteur que l'on divise par 60. Les interrupteurs S1-S4 servent à installer dans le compteur un nombre de départ voulu lorsqu'une impulsion momentanée est appliquée à $\overline{PL}$. Le fonctionnement de la minuterie est lancé en appuyant sur l'interrupteur à bouton-poussoir DÉBUT. La bascule Z sert à éliminer les rebonds de l'interrupteur DÉBUT. Le monostable produit une impulsion très étroite appliquée à l'entrée $\overline{PL}$. La sortie de la bascule X est une forme d'onde qui reste à 1 durant un nombre de secondes égal au nombre installé par les interrupteurs.

a) Supposez que toutes les bascules et que le compteur sont à 0; analysez et expliquez le fonctionnement du circuit, si nécessaire au moyen de formes

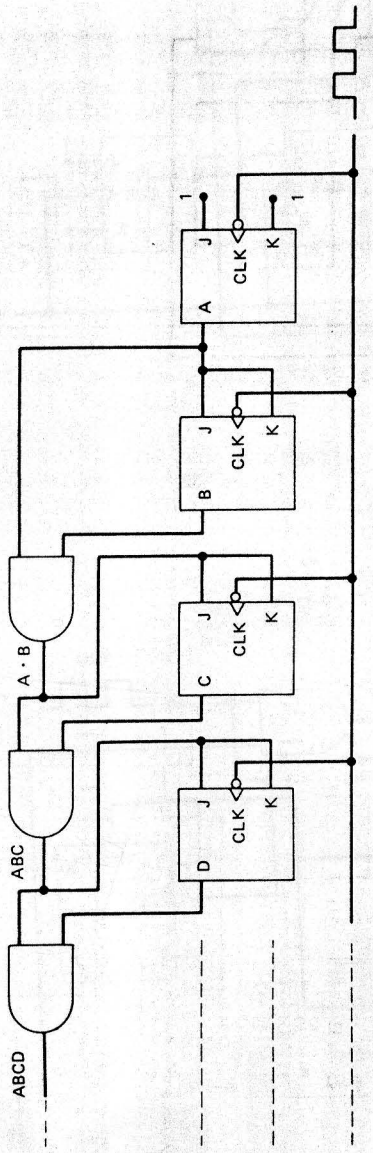


FIGURE 7-60

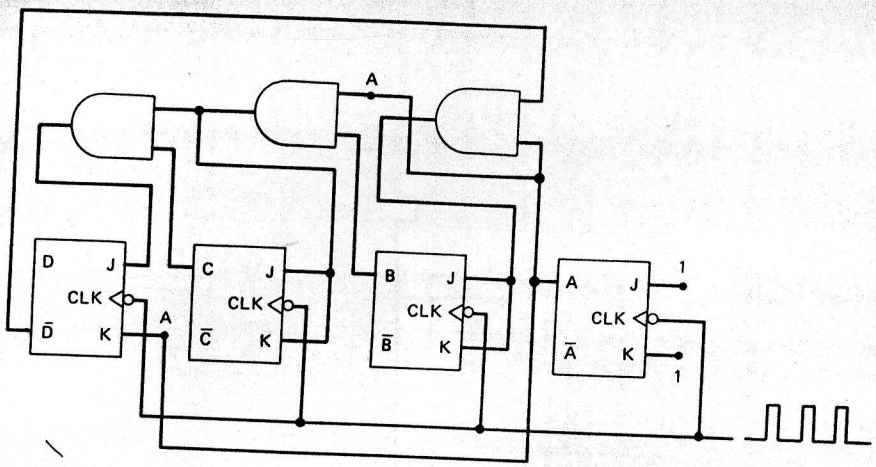
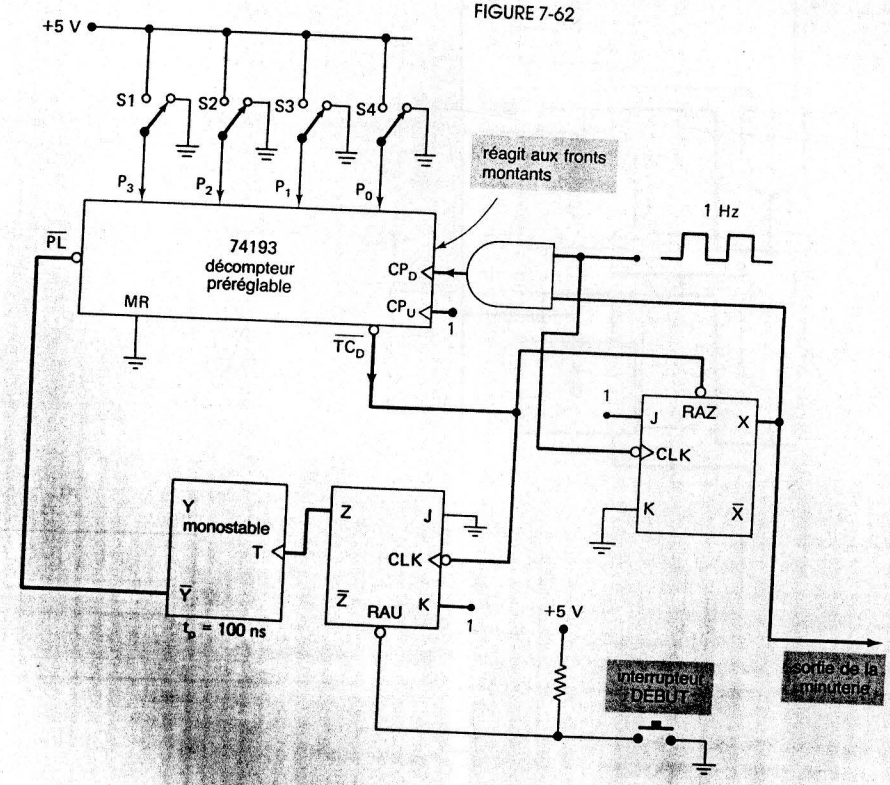


FIGURE 7-61

FIGURE 7-62



d'ondes, pour le cas S1 et S4 = 0 et S2 et S3 = 1. Assurez-vous de bien expliquer le rôle de la bascule X.

b) Dites pourquoi la sortie de la minuterie ne peut être prélevée à la sortie $\overline{TC}_D$.

c) Dites pourquoi on ne peut utiliser l'interrupteur DÉBUT pour déclencher directement le monostable.

d) Dites ce qui se passe si l'interrupteur DÉBUT est enfoncé trop longtemps. Ajoutez les circuits logiques nécessaires qui font en sorte que l'interrupteur DÉBUT enfoncé trop longtemps n'a plus d'effet sur la minuterie.

7-19 Modifiez le circuit de la figure 7-24 pour qu'il fonctionne comme un compteur MODULO-10. La fréquence du signal de sortie Q_3 doit être un dixième de la fréquence du signal d'entrée CP_D. Tracez les formes d'ondes de Q_3, Q_2, Q_1, Q_0 et $\overline{TC}_D$.

7-20 Changez les entrées parallèles de la figure 7-24 pour 1001. Tracez les formes d'ondes de Q_3, Q_2, Q_1, Q_0 et $\overline{TC}_D$. Dites quel est le MODULO.

SECTION 7-10

7-21 La figure 7-63 nous montre le symbole IEEE/ANSI d'un compteur intégré 7490 ou 74290. Examinez bien ce symbole et déduisez les informations suivantes:

a) Le MODULO total

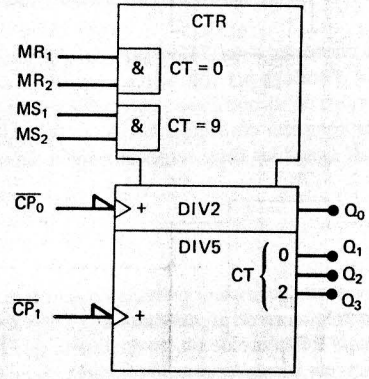
b) Le rôle joué par les entrées *MR*

c) Le rôle joué par les entrées *MS*

d) S'agit-il d'un compteur ou d'un décompteur?

e) Comment faut-il le connecter pour obtenir un compteur DCB (reportez-vous à la fiche technique, Appendice II).

f) Comment faut-il le connecter pour diviser la fréquence d'horloge par 10 et obtenir une onde carrée symétrique.



7490/74290

FIGURE 7-63

7-22 Le compteur intégré 74192 fonctionne exactement comme le 73193 sauf pour les différences que voici:

- Le 74192 est un compteur DCB qui compte aussi bien de 0 à 9 que de 9 à 0.

- La sortie $\overline{TC}_U$ est activée quand le nombre dans le compteur est 9 et que l'entrée CP_U est BASSE.

Modifiez le symbole IEEE/ANSI de la figure 7-25 afin de représenter un 74192.

SECTIONS 7-11 et 7-12

7-23 Dessinez les portes nécessaires au décodage de tous les états d'un compteur MODULO-16 dont les sorties sont vraies au niveau BAS.

7-24 Dessinez les portes ET nécessaires pour décoder les 10 états du compteur DCB de la figure 7-6 b).

7-25 La figure 7-64 reproduit un compteur servant à générer des formes d'ondes de commande. Les formes d'ondes de commande 1 et 2 peuvent commander plusieurs dispositifs, comme des moteurs, des solénoïdes, des soupapes et des appareils de chauffage. Déterminez les formes d'ondes de commande en supposant qu'au départ toutes les bascules sont à 0. Ne tenez pas compte des signaux transitoires de décodage. Supposez que la fréquence d'horloge est 1 kHz.

7-26 Tracez les formes d'ondes de sortie complètes des portes de décodage d'un compteur à *propagation* MODULO-16, en indiquant les transitoires ou décrochements qui peuvent survenir en raison des retards introduits par les bascules. Dites pourquoi les seules portes ayant des transitoires sont celles des nombres *pairs*.

7-27 Le circuit de la figure 7-64 peut fonctionner de façon impropre en raison des transitoires qui apparaissent à la sortie des portes de décodage ET.

a) Déterminez à quels points les signaux transitoires peuvent perturber le fonctionnement.

b) Décrivez les deux façons d'éliminer la possibilité d'un fonctionnement erratique.

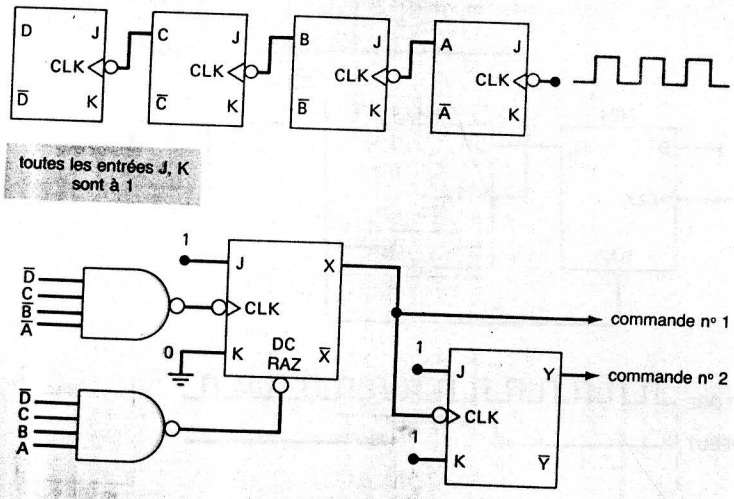


FIGURE 7-64

SECTION 7-13

7-28 Dites combien on utilise de bascules dans la figure 7-31. Indiquez dans quel état se trouve chaque bascule après 795 impulsions.

7-29 Dites combien on doit monter en cascade de compteurs DCB pour compter jusqu'à 8000. Pour cela combien faut-il de bascules? Comparez le nombre de bascules ainsi trouvé au nombre de bascules nécessaire pour compter jusqu'à 8000 avec un compteur binaire ordinaire. Étant donné qu'il faut plus de bascules, dites pourquoi on a quand même recours à des compteurs DCB en cascade.

SECTION 7-14

7-30 a) Concevez un compteur synchrone qui dénombre la séquence suivante: 000, 010, 101, 110 etc. Les états indésirables (qui ne servent pas) sont 001, 011, 100 et 111 et ont toujours comme état suivant 000 après le signal d'horloge suivant.

b) Reprenez la conception du compteur de a), mais cette fois-ci sans tenir compte d'une exigence quelconque pour l'état suivant des états indésirables, c'est-à-dire les états SUIVANTS peuvent être des états indifférents. Comparez votre solution à celle trouvée en a).

7-31 Mettez en pratique la procédure de conception d'un compteur synchrone pour réaliser un décompteur synchrone de 4 bits qui dénombre tous les états de 1111 à 0000. Comparez votre solution à celle du décompteur synchrone décrit à la section 7-7.

SECTION 7-15

7-32 Tracez le schéma d'un compteur en anneau de 5 bits formé de bascules J-K.

7-33 Combinez le compteur à anneau du problème 7-32 à *une* bascule J-K pour réaliser un compteur MODULO-10. Déterminez la suite des états dénombrés par ce compteur. Il s'agit ici d'un exemple de compteur à décade qui n'est pas un compteur DCB.

7-34 Tracez le schéma d'un compteur Johnson MODULO-10 formé de bascules J-K et déterminez la suite des nombres comptés. Tracez le circuit de décodage nécessaire pour chacun des 10 états. Il s'agit là d'un autre exemple de compteur à décade qui n'est pas un compteur DCB.

7-35 Trouvez la fréquence d'impulsion aux points *w*, *x*, *y* et *z* du circuit de la figure 7-65.

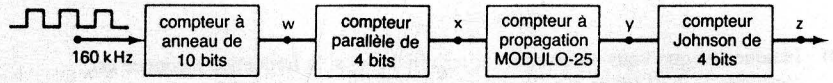


FIGURE 7-65

SECTION 7-16

7-36 Comme nous le mentionnions dans le texte, le compteur de fréquence de la figure 7-51 présente l'inconvénient d'avoir un afficheur qui expose toutes les opérations du compteur (remise à zéro, comptage et maintien), ce qui dérange et peut rendre la lecture impossible. Pour remédier à ce problème, on peut ajouter des registres tampons pour stocker le contenu du compteur à la fin de chaque intervalle de comptage (t_3 à t_4 de la figure 7-41) et conserver cette valeur jusqu'à la fin du prochain intervalle de comptage (t_7 à t_8). La figure 7-66 illustre ce compteur modifié. Un registre formé de quatre bascules D a été inséré entre chaque compteur DCB et l'unité décodeur/afficheur.

a) Analysez ce circuit et expliquez son fonctionnement, particulièrement le transfert des données des compteurs aux afficheurs.

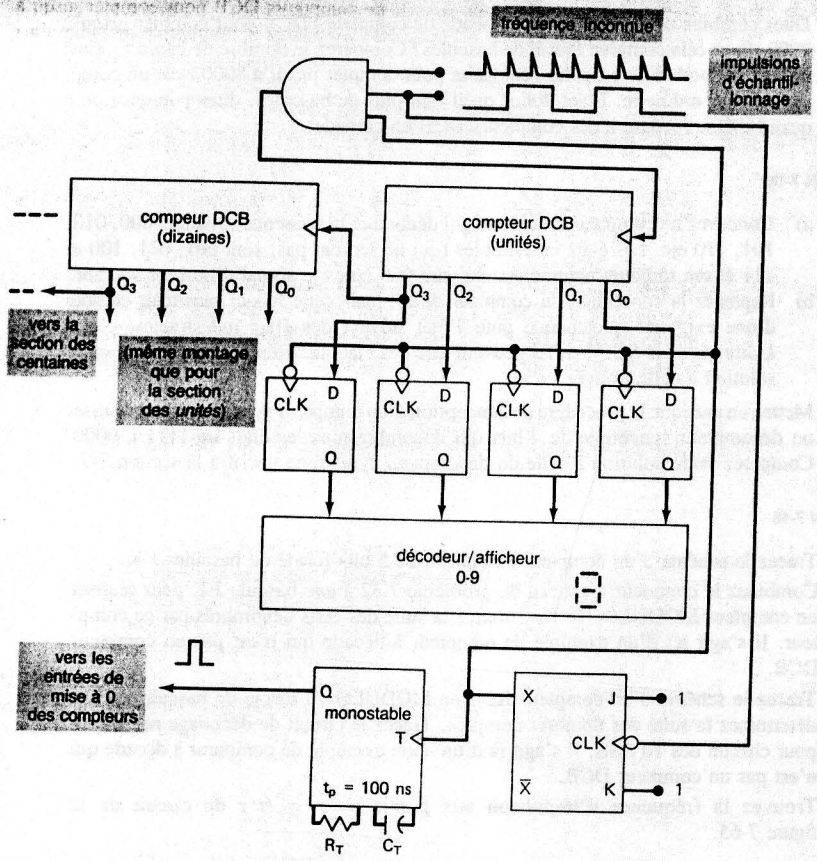


FIGURE 7-66

b) Indiquez ce que vous observerez sur les afficheurs si la fréquence inconnue est constante à 2570 ips et que l'intervalle d'échantillonnage est 0,1 s.

c) Quelle valeur sera affichée si la fréquence inconnue passe soudainement à 3230 ips?

7-37 Le compteur de fréquence de la figure 7-66 a recours à 3 compteurs DCB et à un intervalle d'échantillonnage de 100 μs. Déterminez les nombres apparaissant dans les trois afficheurs de ces compteurs de fréquence pour les fréquences d'entrée suivantes: a) 220 kHz; b) 4,5 Mhz; c) 750 Hz.

SECTION 7-17

7-38 Dessinez le circuit complet de la section des SECONDES de l'horloge numérique de la figure 7-42. Utilisez une puce 74293 pour le compteur MODULO-6 et une puce 74293 pour le compteur DCB.

7-39 L'horloge numérique de la figure 7-42 doit être munie d'un certain dispositif manuel pour régler les HEURES et les MINUTES. Une façon de faire consiste à dériver le signal d'une impulsion par seconde sur l'entrée de la section des MINUTES quand on appuie sur un BOUTON-POUSSOIR DE RÉGLAGE DES MINUTES. Et de même quand on appuie sur un BOUTON-POUSSOIR DE RÉGLAGE DES HEURES. Concevez le circuit logique qui permet de régler l'horloge au moyen de deux interrupteurs boutons-poussoir.

7-40 Modifiez la section des HEURES de l'horloge numérique de la figure 7-43 pour qu'elle compte et affiche les heures selon le format métrique (de 00 à 23 heures).

7-41 Modifiez le compteur de fréquence de la figure 7-66 pour employer des CI 74174. Supposez que ce compteur contient trois compteurs DCB et un afficheur de 3 bits.

7-42 Montrez comment connecter deux 74178 pour réaliser un compteur en anneau de 8 bits.

7-43 La figure 7-67 montre comment un 74178 peut être utilisé comme *convertisseur parallèle-série*. Les données parallèles qui sont introduites sur P_0–P_3 sont décalées les unes après les autres de sorte qu'une forme d'onde série apparaît en Q_3. Le décalage des données parallèles est contrôlé par l'arrivée d'une impulsion DÉBUT. Cette impulsion DÉBUT ne se produit pas en même temps que les impulsions d'horloge; les bascules *D* servent donc à synchroniser le chargement des données parallèles et le décalage dans la puce 74178.

Supposez que l'impulsion DÉBUT soit demeurée à 0 et que les impulsions d'horloge n'aient cessé d'arriver au circuit avant t_0 (voir formes d'ondes). Tracez les formes d'ondes qui apparaissent aux bornes Q_X, $\overline{Q}_X$, Q_Y et Q_3 après l'application de l'impulsion DÉBUT illustrée sur la figure.

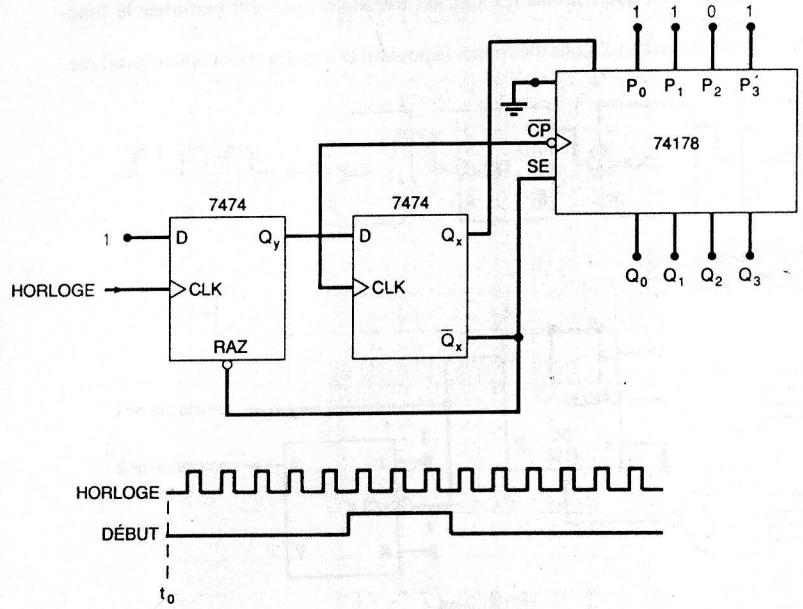


FIGURE 7-67

7-44 On utilise souvent un registre à décalage pour retarder un signal logique d'un nombre entier de cycles d'horloge. Si les formes d'ondes de la figure 7-68 sont appliquées aux entrées d'un des registre à décalage de la puce 4731B, dites combien de temps

prendra D_S avant d'apparaître sur la sortie Q_{63}. (Supposez que la forme d'onde D respecte l'exigence t_M des bascules.)

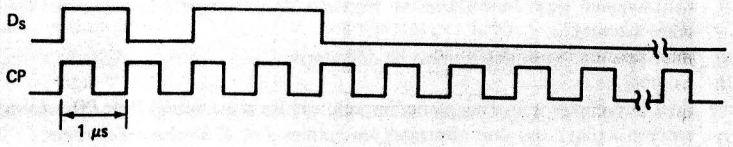


FIGURE 7-68

SECTIONS 7-21 et 7-22

7-45 Modifiez le circuit de la figure 7-52 afin que la sortie de l'INVERSEUR soit connectée à l'entrée A au lieu de $\overline{MR}$.

a) Dessinez les formes d'ondes de chaque sortie de bascule produites en réponse aux formes d'ondes d'entrée illustrées à la figure 7-69.

b) Ajoutez les portes logiques nécessaires pour produire un signal de synchronisation qui passe au niveau HAUT seulement durant les intervalles t_1 à t_2 et t_8 à t_9.

c) Ajoutez les portes logiques nécessaires pour produire un signal de synchronisation qui passe au niveau BAS pendant l'intervalle t_4 à t_7.

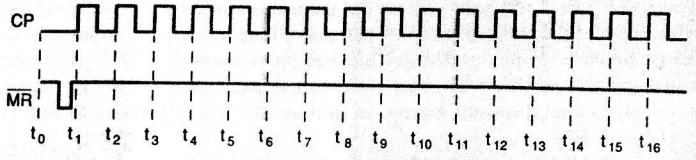
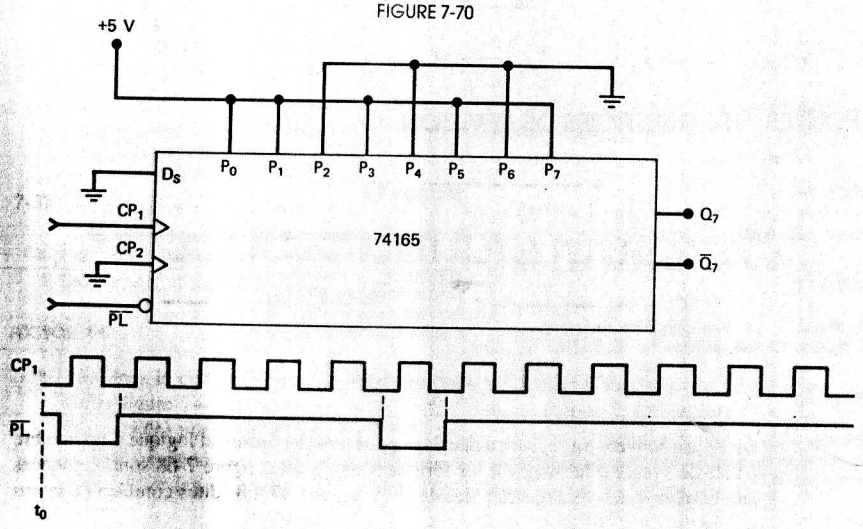


FIGURE 7-69

7-46 Supposez que les impulsions sont appliquées continuellement sur CP_1 avant t_0 sur la figure 7-70. Dessinez la forme d'onde en Q_7 qui constitue la réponse aux formes d'ondes CP_1 et $\overline{PL}$ illustrées.

FIGURE 7-70



7-47 Pendant qu'il étudie le schéma d'une pièce d'appareil, un technicien ou un ingénieur rencontre un circuit intégré qu'il ne connaît pas. Dans de tels cas, il est souvent indispensable de consulter le recueil des fiches techniques du fabricant pour connaître les spécifications de l'élément. Les renseignements sur la fiche technique sont toujours complets, quoique parfois difficiles à interpréter, surtout par quelqu'un peu expérimenté. Le problème que voici vise à vous exercer à trouver les renseignements nécessaires au sujet d'un CI relativement complexe — le registre à décalage universel et bidirectionnel 74194. Consultez les fiches techniques de l'Appendice II et répondez aux questions suivantes. Justifiez vos réponses.

a) L'entrée *CLR* est-elle asynchrone ou synchrone?

b) *Vrai ou faux:* Quand *CLK* est BAS, les niveaux de S_0 et S_1 n'ont pas d'effet sur le registre.

c) Supposez vraies les conditions que voici:

$$Q_AQ_BQ_CQ_D = 1\ 0\ 1\ 1$$
$$ABCD = 0\ 1\ 1\ 0$$
$$\overline{CLR} = 1$$
$$\text{SR SER} = 0$$
$$\text{SL SER} = 1$$

Si $S_0 = 0$ et $S_1 = 1$, indiquez les valeurs de sortie du registre après une impulsion d'horloge. Après deux. Après trois. Après quatre.

d) Pour les mêmes conditions et la même question qu'en c), mais cette fois-ci avec $S_0 = 1$ et $S_1 = 0$.

e) Et maintenant pour $S_0 = S_1 = 1$.

f) Et maintenant pour $S_0 = S_1 = 0$.

g) Mêmes conditions qu'en c), mais cette fois-ci la sortie Q_A est connectée à SL SER. Quelles sont les sorties du registre après quatre impulsions *CLK*?

h) Montrez comment connecter ce CI pour qu'il fonctionne comme un compteur en anneau dont la séquence de répétition $Q_AQ_BQ_CQ_D$ est: 0001, 0010, 0100, 1000.

SECTION 7-24

7-48 Un technicien soumet à un test le compteur de la figure 7-54 a) en appliquant un signal d'horloge de basse fréquence et en surveillant les sorties des bascules sur les DEL témoins. Il observe la répétition de la séquence suivante:

Q_3	Q_2	Q_1	Q_0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1

Quelles sont les raisons probables de l'erreur de comptage de ce dispositif.

7-49 Reportez-vous au circuit de l'horloge numérique des figures 7-42 et 7-43. Un technicien constate que les sections des SECONDES et des MINUTES dénombrent correctement ces valeurs, alors que la section des HEURES est fautive et compte

ainsi: 01, 02, 03, 04, 05, 06, 07, 08, 09, 10, 11, 12, 11, 12, 11, 12... Trouvez la cause probable de ce dérèglement.

7-50 Un technicien teste le circuit de l'horloge numérique (figures 7-42 et 7-43), et il observe que la section des HEURES ne dénombre pas correctement cette valeur, tandis que celle des MINUTES compte de 00 à 39, puis revient à zéro et recommence. Quelles sont les causes possibles de cette défectuosité?

7-51 Reportez-vous au fréquencemètre modifié de la figure 7-66. Supposez que les trois étages de compteur DCB possèdent des tampons associés. L'intervalle d'échantillonnage est réglée à 1 s, et la fréquence inconnue est 125 ips. Décrivez ce qu'on peut lire sur l'afficheur en présence de chacune des défaillances du circuit que voici:

a) Une connexion ouverte dans l'entrée du haut de la porte ET.
b) Une résistance R_T grillée.

7-52 Une technicienne teste le fréquencemètre de la figure 7-66 avec un intervalle d'échantillonnage de 1 s et une fréquence inconnue de 125 ips. Elle prévoit observer sur l'afficheur 125, mais relève plutôt les données suivantes: 125, 250, 375, 500, 625, 750, 875, 000, 125, 250,... Quelle est la raison probable d'un tel comportement?

7-53 Reportez-vous à la figure 7-18. Décrivez de quelle façon chacune des défaillances de circuit que voici altère les opérations de comptage ou de décomptage:

a) La sortie 4 de la porte ET est court-circuitée à V_{CC} par l'intérieur.
b) Un pont de soudure court-circuite à la masse la sortie 1 de la porte ET.

7-54 Un technicien procède à un essai de la minuterie de la figure 7-62 et consigne les mesures suivantes:

S1	S2	S3	S4	SORTIE DE LA MINUTERIE
5 V	0 V	0 V	0 V	10
0 V	0 V	5 V	5 V	3
5 V	0 V	5 V	5 V	11
5 V	5 V	5 V	0 V	14
0 V	5 V	0 V	5 V	7
5 V	5 V	0 V	0 V	14

Examinez ces valeurs et déterminez les causes possibles de ce dérèglement.

7-55 Un technicien réalise le câblage du compteur de la figure 7-59. Il applique un signal de précision de 8,64 kips à l'entrée et mesure en X une fréquence de 54 ips au lieu de la valeur prévue 60 de ips. Indiquez quelle erreur de câblage il a sans doute commise.

CONTRÔLE

7-56 Pour chacune des affirmations qui suivent, nommez le type de compteur que l'on caractérise.

a) Chaque bascule est commandée par l'horloge au même moment.
b) Chaque bascule divise la fréquence de son horloge par 2.
c) La séquence de comptage est 111, 110, 101, 100, 011, 010, 001, 000.
d) Le compteur a 10 états distincts.
e) Le retard de commutation total est égal à la somme des retards des bascules.
f) Ce compteur ne nécessite pas de circuits logiques de décodage.
g) Le MODULO est toujours deux fois le nombre de bascules.
h) Ce compteur divise la fréquence d'entrée par son MODULO.
i) Ce compteur peut dénombrer ses états dans un sens ou dans l'autre.
j) Ce compteur peut commencer sa séquence de comptage à la valeur de départ que l'on veut.
k) Peut être altéré par des pointes de décodage.
l) Compte de 0 à 9.
m) Peut être conçu pour compter n'importe quelle séquence si l'on détermine la relation logique qui doit alimenter les entrées J et K de chaque bascule.

APPLICATION AU MICRO-ORDINATEUR

7-57 Un microprocesseur de contrôle doit le plus souvent surveiller la temporisation d'événements extérieurs comme mettre en marche ou arrêter des solénoïdes, des moteurs, ou des relais. De telles commandes peuvent être chronométrées par un logiciel qui répète une boucle un certain nombre de fois. Cette solution sollicite grandement l'UCT qui pendant le calcul de la boucle ne peut faire rien d'autre. C'est pourquoi, la plupart des durées de temporisation sont confiées à du matériel *sous le contrôle de l'UCT*. Autrement dit, l'UCT transfère au matériel des données qui informent ce dernier de la durée de la temporisation à réaliser.

Au problème 7-18, nous avons vu comment le CI 74193 pouvait servir dans une minuterie (figure 7-62) pour générer des durées de temporisation précises correspondant aux données binaires définies par quatre interrupteurs. Il est possible de modifier ce circuit pour que les données binaires proviennent de l'UCT plutôt que d'interrupteurs. À la section 5-20, nous avons vu comment une UCT pouvait transférer des données à un dispositif externe en utilisant ses sorties adresse, donnée et horloge (figure 5-50).

Montrez comment on peut combiner ces deux circuits pour que la sortie de la minuterie reste au niveau HAUT pendant un intervalle de temps égal au nombre binaire avec lequel l'UCT a initialisé le compteur 74193. Vous pouvez éliminer tous les circuits inutiles. Supposez que le signal *CP* de l'UCT est une onde carrée de 1 MHz. Rappelez-vous que $\overline{PL}$ est une entrée asynchrone.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 7-1

1. Faux **2.** 0000

SECTION 7-2

1. *D*, *C* et *A*. **2.** Vrai, puisqu'un compteur DCB possède 10 états distincts. **3.** 5 kHz.

SECTION 7-3

1. 250 Hz. **2.** $f_{en}/60$. **3.** 4096. **4.** Ce compteur est à MODULO-64 et divise la fréquence par 64.

SECTION 7-4

1. Dans un compteur, le dénombrement progresse de 1 à chaque impulsion d'horloge; dans un décompteur, le dénombrement régresse de 1 à chaque impulsion d'horloge. **2.** La sortie inversée de chaque bascule est connectée à l'entrée *CLK* de la bascule suivante.

SECTION 7-5

1. Chaque bascule additionne son retard de propagation au retard total de réaction du compteur à une impulsion d'horloge. **2.** MODULO-256.

SECTION 7-6

1. Peut fonctionner à des fréquences supérieures et contient des circuits plus complexes.
2. Six bascules et quatre portes ET.

SECTION 7-8

1. Ils peuvent être initialisés avec n'importe quelle valeur de départ. **2.** L'initialisation asynchrone est indépendante du signal d'horloge, tandis que l'initialisation synchrone se manifeste à l'instant d'un front actif du signal d'horloge.

SECTION 7-9

1. Quand $\overline{PL}$ est forcée momentanément au niveau BAS, le compteur est réinitialisé avec le nombre binaire se trouvant sur les entrées P_0 à P_3.
2. Un niveau HAUT sur *MR* a la priorité sur toutes les autres entrées et réinitialise le compteur à 0000. **3.** Vrai. **4.** 1, 1, 0 respectivement.

SECTION 7-10

1. Voyez le texte qui s'y rapporte.
2. a) Fonctionnement en compteur, **b)** cette entrée est multipliée logiquement (ET) avec une autre entrée ou sortie possédant un « 4 » dans son libellé; **c)** cette entrée a le contrôle sur l'effet de toute entrée ayant un « 5 » dans son libellé; **d)** l'entrée de donnée contrôlée par l'entrée libellée C5.

SECTION 7-11

1. Une porte NON-ET à 6 entrées identifiée *A*, *B*, *C*, $\overline{D}$, *E* et $\overline{F}$.

SECTION 7-12

1. Les décrochements de tension seront causés par le changement des états des bascules une à la fois pendant les transitions d'état du compteur.
2. Le signal de strobage bloque les portes de décodage tant que toutes les bascules n'ont pas terminé leur transition.

SECTION 7-14

1. Voyez le texte. **2.** Les niveaux nécessaires qui doivent se trouver sur les entrées *J* et *K* pour produire l'ensemble des transitions d'état des bascules. **3.** Celle des niveaux nécessaires sur les entrées *J* et *K* pour réaliser les transitions d'état du compteur. **4.** Vrai.

SECTION 7-15

1. Compteur en anneau. **2.** Compteur Johnson. **3.** La sortie complémentée de la dernière bascule est rétrocouplée sur l'entrée de la première bascule. **4.** Faux, vrai, vrai.

SECTION 7-16

1. 1 ms. **2.** Compteur mis à zéro; compteur dénombre des impulsions durant l'intervalle d'échantillonnage; compteur s'arrête et conserve le nombre en vue de son affichage.

SECTION 7-17

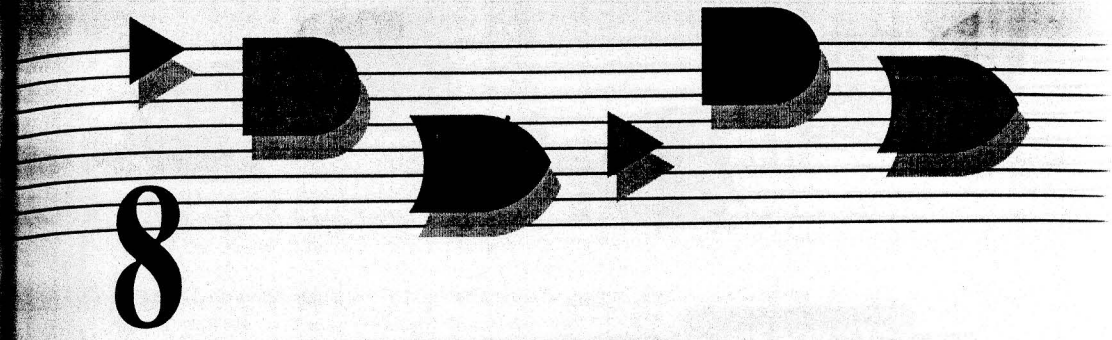
1. Conformateur d'impulsions, diviseur de fréquence, compteur des secondes et afficheur, compteur des minutes et afficheur, compteur des heures et afficheur. **2.** Pour transformer le FD provenant de la section des MINUTES en un FM nécessaire au fonctionnement du 74192.

SECTION 7-18 à la SECTION 7-22

1. Écriture parallèle/lecture série. **2.** Vrai. **3.** Écriture série/lecture parallèle. **4.** Écriture série/lecture série.

SECTION 7-23

1. Elle sépare les deux indications de fonction de l'entrée. **2.** SRG 64.

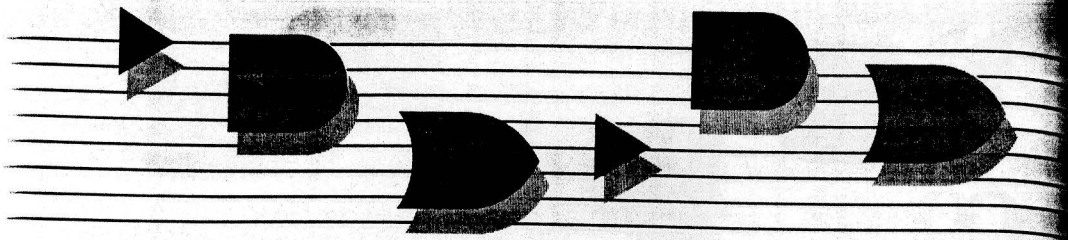


8

FAMILLES DE CIRCUITS INTÉGRÉS LOGIQUES

Contenu

8.1 Terminologie des CI numériques
8.2 Famille logique TTL
8.3 Caractéristiques de la série TTL standard
8.4 Autres séries TTL
8.5 Facteurs de charge et sortance des TTL
8.6 Autres caractéristiques TTL
8.7 Sorties TTL à collecteur ouvert
8.8 TTL trois états
8.9 Famille de CI numériques ECL
8.10 Circuits intégrés numériques MOS
8.11 Transistor MOS
8.12 Circuits numériques MOS
8.13 Caractéristiques de la logique MOS
8.14 Logique MOS complémentaire
8.15 Caractéristiques des séries CMOS
8.16 CMOS à drain ouvert et sorties trois états
8.17 Porte de transmission CMOS (interrupteur bilatéral)
8.18 Interfaçage des CI
8.19 CMOS piloté par TTL
8.20 TTL piloté par CMOS
8.21 Dépannage



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Lire les fiches techniques des fabricants et comprendre la terminologie employée.
- Comparer les caractéristiques de la série TTL standard à celles des autres séries TTL.
- Déterminer la sortance d'un certain dispositif logique.
- Exploiter des éléments logiques à collecteur ouvert pour réaliser des ET câblés.
- Analyser des circuits renfermant des dispositifs trois états.
- Décrire les principaux caractères des familles logiques TTL, ECL, MOS et CMOS, ainsi que leurs différences.
- Indiquer les principales considérations qui interviennent quand on cherche à joindre (interfacer) des éléments de familles logiques différentes.
- Employer un pulseur logique et un traceur galvanométrique comme outils de dépannage.

Introduction

Comme nous l'avons décrit au chapitre 4, la technologie des circuits intégrés a fait des bonds prodigieux pendant sa transition de l'intégration à petite échelle (SSI) qui loge moins de 12 portes par puce, à l'intégration à ultragrande échelle (ULSI) qui parvient à placer jusqu'à 100 000 portes sur une seule puce, et pendant les étapes intermédiaires de l'intégration à moyenne échelle (12 à 99 portes) et de l'intégration à grande et à très grande échelle (des dizaines de milliers de portes par puce).

La plupart des raisons qui expliquent la prépondérance des circuits intégrés dans les systèmes numériques sont évidentes. D'abord, un CI concentre un grand nombre de circuits dans un petit espace, d'où une réduction proportionnelle des dimensions de tous les systèmes numériques. Ensuite, diminution considérable des coûts en raison des économies générées par la production en grande série de dispositifs identiques. Mais à ces raisons, il faut en ajouter d'autres, moins évidentes.

Les circuits intégrés rendent les systèmes numériques plus fiables parce qu'ils diminuent le nombre d'interconnexions entre éléments. Avant les circuits intégrés, chaque connexion reliait un composant discret (transistor, diode, résistance) à un autre composant. Maintenant, la majeure partie des connexions se trouvent à l'intérieur d'un boîtier, à l'abri d'un soudage peu soigné, de coupures ou de bris des conducteurs de la plaquette, et d'autres agressions physiques. Les CI ont abaissé dans une proportion phénoménale la consommation d'énergie électrique pour une fonction donnée, puisque les circuits miniatures requièrent beaucoup moins de courant que leur équivalent en composants discrets. À cette réduction de la consommation, il faut ajouter les économies réalisées par un besoin de ventilation moins important.

Cependant, certaines choses ne sont pas à la portée des CI: ils ne soutiennent pas de forts courants et de fortes tensions, car la chaleur dégagée par ces derniers serait suffisante pour porter ces petits blocs à des températures destructrices. En outre, les CI ne se prêtent pas particulièrement bien à l'implantation de certains composants électriques comme les bobines, les transformateurs et les condensateurs de grande capacité. Pour toutes ces raisons, les CI sont réservés aux opérations de circuits basse puissance, celles qui effectuent ce qu'on appelle le *traitement de l'information*. Les opérations qui doivent avoir recours à des dispositifs non intégrables ou à des niveaux de haute puissance restent matérialisées par des composants discrets.

La diffusion à grande échelle des circuits intégrés rend désormais indispensable la connaissance et la compréhension des caractéristiques électriques de la plupart des grandes familles de circuits logiques intégrés. Rappelons-nous que ce qui différencie les diverses familles, ce sont les éléments utilisés dans leurs circuits. Par exemple, dans les familles TTL et ECL on utilise le transistor bipolaire comme élément de base; dans les familles PMOS, NMOS et CMOS, c'est le transistor MOS qui sert de bloc élémentaire. Le présent chapitre expose les caractéristiques importantes de chacune des familles et des sous-familles de CI. Une fois la maîtrise de ces caractéristiques acquise, vous serez plus apte à analyser, à dépanner et à concevoir des circuits numériques regroupant des éléments de différentes familles.

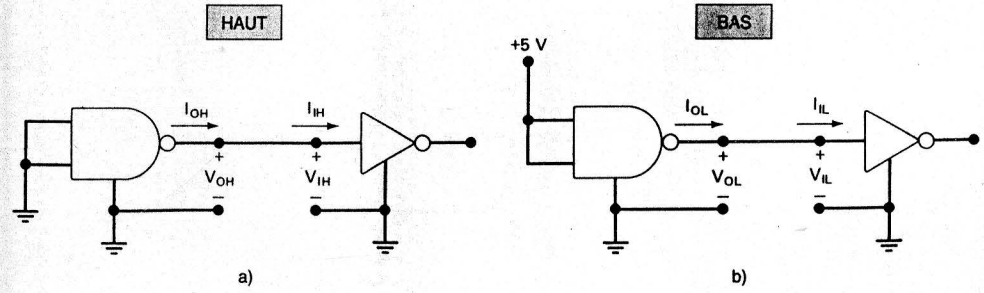
8-1 TERMINOLOGIE DES CI NUMÉRIQUES

Malgré le grand nombre de fabricants de CI intégrés, la terminologie utilisée est heureusement presque normalisée. Voici la définition et l'explication des termes les plus courants.

Paramètres de courants et de tensions (voir figure 8-1)

V_{IH}(min) — tension d'entrée niveau HAUT: le niveau de tension nécessaire pour avoir un 1 logique en *entrée*. Toute tension inférieure à ce niveau n'est pas considérée comme un état HAUT par le circuit logique.

FIGURE 8-1 Courants et tensions dans les deux états logiques.



V_{IL}(max) — tension d'entrée niveau BAS: le niveau de tension nécessaire pour avoir un 0 logique en *entrée*. Toute tension supérieure à ce niveau n'est pas considérée comme un état BAS par le circuit logique.
V_{OH}(min) — tension de sortie niveau HAUT: le niveau de tension de la *sortie* d'un circuit logique correspondant à l'état logique 1. La valeur minimale de V_{OH} est généralement spécifiée.
V_{OL}(max) — tension de sortie niveau BAS: le niveau de tension à la *sortie* d'un circuit logique correspondant à l'état logique 0. La valeur maximale de V_{OL} est généralement spécifiée.
I_{IH} — courant d'entrée niveau HAUT: le courant qui traverse une borne d'entrée quand une tension niveau haut est appliquée à cette entrée.
I_{IL} — courant d'entrée niveau BAS: le courant qui traverse une entrée quand une tension niveau bas est appliquée à cette entrée.
I_{OH} — courant de sortie niveau HAUT: le courant qui traverse une borne de sortie placée au niveau logique 1 dans des conditions de charge spécifiées.
I_{OL} — courant de sortie niveau BAS: le courant qui traverse une borne de sortie placée au niveau logique 0 dans des conditions de charge spécifiées.

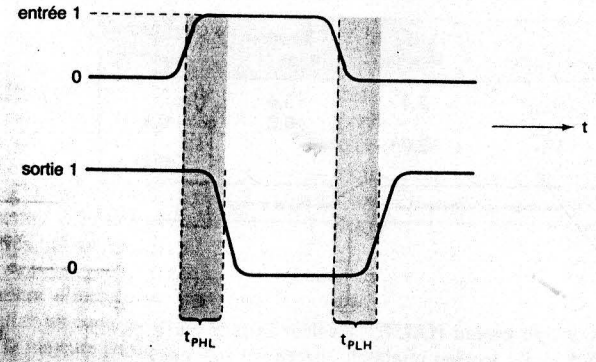
Sortance Normalement, la sortie d'un circuit logique doit piloter plusieurs entrées logiques. La *sortance* (appelée également *facteur de charge*) est définie comme le nombre *maximal* d'entrées logiques standard qui peuvent être pilotées sans problèmes par une sortie. Par exemple, quand il est indiqué qu'une porte logique a une sortance de 10, cela signifie qu'elle peut piloter 10 entrées logiques standard. Si on dépasse ce nombre, il n'est pas assuré que les tensions des niveaux logiques des sorties seront exactes.

Retards de propagation Un signal logique qui traverse un circuit subit toujours un retard. Deux retards de propagation sont définis:

t_{PLH}: retard pour passer du niveau logique 0 au niveau logique 1 (BAS à HAUT).
t_{PHL}: retard pour passer du niveau logique 1 au niveau logique 0 (HAUT à BAS).

La figure 8-2 représente les retards de propagation d'un INVERSEUR. Notez que t_{PHL} est le retard de la réponse quand la sortie passe du niveau HAUT au niveau BAS. Ce retard est mesuré entre les points à mi-hauteur des transitions d'entrée et de sortie. La valeur t_{PLH} est le retard de la réponse quand la sortie passe du niveau BAS au niveau HAUT.

FIGURE 8-2 Retards de propagation d'un inverseur.



Généralement, t_{PHL} et t_{PLH} sont différents et varient tous les deux selon les conditions de charge. Les temps de propagation servent généralement de mesure pour la vitesse relative des circuits logiques. Par exemple, un circuit logique dont les temps sont 10 ns est plus rapide qu'un circuit dont les temps sont 20 ns.

Consommation Tous les CI, quand ils sont en fonctionnement, consomment une certaine énergie électrique. Cette énergie est délivrée par une ou plusieurs tensions d'alimentation raccordées aux broches d'alimentation du boîtier. Généralement, il n'y a qu'une borne d'alimentation par boîtier, appelée V_{CC} (pour les TTL) ou V_{DD} (pour les MOS).

La puissance dissipée par un CI est généralement spécifiée par le courant I_{CC} qu'il tire de l'alimentation V_{CC}, de sorte que la puissance réelle est le produit $I_{CC} \times V_{CC}$. Dans le cas de nombreux CI, le courant qu'ils drainent de l'alimentation varie en fonction des états logiques des circuits de la puce. Par exemple, la figure 8-3 a) nous fait voir une puce NON-ET dont *toutes les sorties* des portes sont au niveau HAUT. Le courant tiré de l'alimentation V_{CC} dans ce cas est appelé I_{CCH}. Par contre, la figure 8-3 b) représente le courant drainé quand *toutes les sorties* des portes sont au niveau BAS. Ce dernier courant est appelé I_{CCL}.

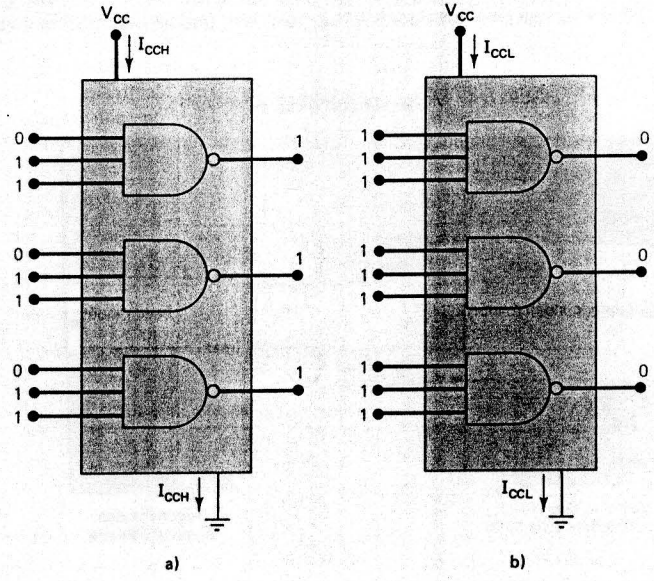
Habituellement, I_{CCH} et I_{CCL} ne sont pas égaux. Le courant moyen est

$$I_{CC}(\text{moy}) = \frac{I_{CCH} + I_{CCL}}{2}$$

et c'est cette valeur qu'on utilise pour calculer la puissance moyenne consommée:

$$P_D(\text{moy}) = I_{CC}(\text{moy}) \times V_{CC}$$

FIGURE 8-3 I_{CCH} et I_{CCL}.



Produit vitesse-consommation Les familles de circuits intégrés ont toujours été caractérisées à la fois par la vitesse et la consommation. Généralement, il est souhaitable d'avoir un délai de propagation pour la porte de même qu'une consommation d'énergie les plus faibles possibles. Comme nous le verrons ci-après, les diverses familles et sous-familles de circuits logiques ont des valeurs pour la vitesse et la consommation très variables qui se répartissent dans un spectre très large. Une façon commode de mesurer la performance globale d'une famille est le calcul du *produit vitesse-consommation* que l'on obtient en multipliant le retard de propagation d'une porte par la consommation d'énergie de celle-ci. Par exemple, supposons une famille dont le retard moyen est 10 ns et la consommation moyenne, 5 mW. Le produit vitesse-consommation est donc:

$$10 \text{ ns} \times 5 \text{ mW} = 50 \times 10^{-12} \text{ watt-seconde}$$
$$= 50 \text{ picojoules (pj)}$$

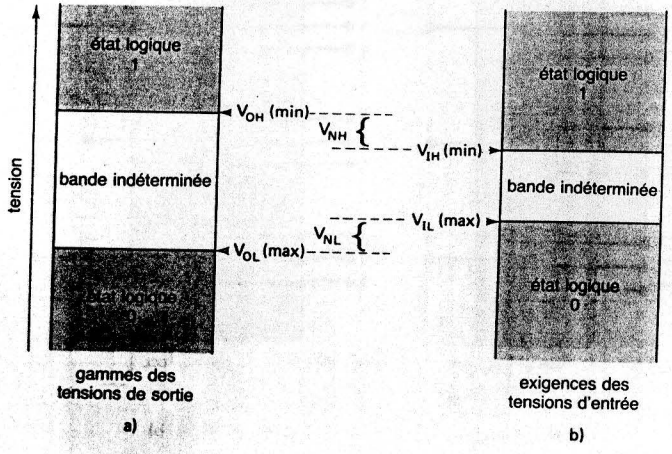
Remarquez que lorsque le retard de propagation est en nanosecondes et que la consommation est en milliwatts, le produit vitesse-consommation est exprimé en picojoules.

Comme on s'en doute bien, un produit vitesse-consommation faible est le but recherché. Les concepteurs de circuits intégrés conjuguent leurs efforts pour abaisser ce produit, soit en augmentant la vitesse du CI (en réduisant son retard de propagation), soit en diminuant la consommation d'énergie. La nature des circuits de commutation à transistors étant ce qu'elle est, il est difficile d'obtenir les deux simultanément.

Immunité aux bruits Les champs électriques et magnétiques parasites peuvent induire des tensions dans les fils de raccordement des circuits logiques. Ces signaux indésirables sont appelés *bruits* et peuvent parfois amener la tension sur la borne d'un circuit logique sous la valeur V_{IH}(min) ou la porter au-dessus de V_{IL}(max) et ainsi perturber le fonctionnement normal. L'*immunité aux bruits* d'un circuit logique définit l'aptitude du circuit à tolérer des tensions parasites sur ses entrées. La mesure quantitative de l'immunité aux bruits est appelée la **marge de sensibilité aux bruits** et elle est illustrée à la figure 8-4.

La figure 8-4 a) schématise la gamme des tensions qui apparaissent à la sortie d'un circuit logique. Toutes les tensions supérieures à V_{OH}(min) sont traduites comme un niveau

FIGURE 8-4 Marges de sensibilité aux bruits CC.



1 logique, et toutes les tensions inférieures à V_{OL}(max) sont traduites comme un niveau logique 0. Les tensions qui se trouvent dans la bande indéterminée ne doivent jamais apparaître sur la borne de sortie d'un circuit logique dans des conditions normales. Les tensions représentées sur la figure 8-4 b) sont celles qu'il faut respecter sur l'entrée d'un circuit logique. Un circuit logique en réponse à toutes les tensions supérieures à V_{IH}(min) se mettra dans l'état logique 1; et il se mettra dans l'état logique 0 en réponse à toutes les tensions inférieures à V_{IL}(max). Les tensions se trouvant dans la bande indéterminée occasionnent une réponse imprévisible et ne doivent jamais être utilisées.

La **marge de sensibilité aux bruits état HAUT** V_{NH} est définie comme suit:

$$V_{NH} = V_{OH}(\text{min}) - V_{IH}(\text{min}) \tag{8-1}$$

et elle est illustrée sur la figure 8-4. V_{NH} est la différence entre la sortie HAUTE la plus basse possible et la tension d'entrée minimale nécessaire pour produire un niveau HAUT. Quand une sortie logique HAUTE attaque l'entrée d'un circuit logique, toute pointe négative supérieure à V_{NH} qui vient parasiter la ligne du signal provoque un abaissement de la tension qui se retrouve ainsi dans la bande indéterminée, là où le fonctionnement du circuit devient imprévisible.

La **marge de sensibilité aux bruits état BAS** V_{NL} est définie comme suit:

$$V_{NL} = V_{IL}(\text{max}) - V_{OL}(\text{max}) \tag{8-2}$$

et c'est la différence entre la sortie BASSE la plus élevée possible et la tension d'entrée maximale nécessaire pour donner lieu à un niveau BAS. Quand une sortie logique BASSE attaque une entrée logique, toute pointe positive supérieure à V_{NL} qui parasite la ligne du signal provoque l'élévation de la tension qui se retrouve ainsi dans la bande indéterminée.

EXEMPLE 8-1

On donne ci-après le tableau des spécifications des tensions d'entrée et de sortie de la famille TTL standard. Utilisez ces valeurs pour déterminer:

a) L'amplitude maximale du signal parasite qui est tolérable quand une sortie au niveau HAUT pilote une entrée.

b) L'amplitude maximale du signal parasite tolérable quand une sortie au niveau BAS pilote une entrée.

Paramètre	Min (*V*)	Tension type (*V*)	Max (*V*)
V_{OH}	2,4	3,6	
V_{OL}		0,2	0,4
V_{IH}	2,0*		
V_{IL}			0,8*

* On ne donne normalement que le minimum V_{IH} et le maximum V_{IL}.

Solution

a) Quand la sortie est au niveau HAUT, la valeur la plus basse qu'elle peut avoir est V_{OH}(min) = 2,4 V. La tension minimale placée sur une entrée faisant en sorte que

cette entrée est au niveau HAUT est $V_{IH}(\text{min}) = 2{,}0$. L'amplitude du signal parasite qui amène la tension sous 2,0 V doit être supérieure à:

$$V_{NH} = V_{OH}(\text{min}) - V_{IH}(\text{min})$$
$$= 2{,}4\text{ V} - 2{,}0\text{ V} = 0{,}4\text{ V}$$

b) Quand une sortie est au niveau BAS, la tension la plus élevée qu'on peut trouver sur cette borne est $V_{OL}(\text{max}) = 0{,}4$ V. La tension maximale que l'on peut placer sur une entrée et qui laisse cette entrée au niveau BAS est $V_{IL}(\text{max}) = 0{,}8$ V. L'amplitude d'un signal parasite positif qui amène cette tension au-delà de 0,8 V doit être supérieure à

$$V_{NL} = V_{IL}(\text{max}) - V_{OL}(\text{max})$$
$$= 0{,}8\text{ V} - 0{,}4\text{ V} = 0{,}4\text{ V}$$

Marge de sensibilité aux bruits CA Plus précisément, les marges de sensibilité aux bruits prévues par les expressions (8-1) et (8-2) sont appelées marges de sensibilité aux bruits CC. Il peut sembler assez contradictoire de donner le qualificatif de courant continu à ces marges, puisque le bruit par définition est généralement un signal alternatif transitoire. Cependant, dans les circuits intégrés rapides qui sont fabriqués de nos jours, une impulsion d'une durée de 1 μs est extrêmement longue et peut être considérée comme du courant continu par rapport à la réponse du circuit logique. À mesure que les durées d'impulsion diminuent pour atteindre la gamme de quelques nanosecondes, on atteint une limite où *la durée d'impulsion* est trop courte pour que le circuit puisse répondre. À ce point, l'amplitude de l'impulsion doit augmenter considérablement pour se répercuter sur la sortie du circuit. Tout cela signifie donc qu'un circuit logique peut tolérer des parasites d'une amplitude très importante si la durée de l'impulsion est très courte comparée au temps de réponse du circuit (retards de propagation). Autrement dit, les *marges de sensibilité aux bruits CA* sont habituellement beaucoup plus intenses que les marges CC données par les expressions (8-1) et (8-2).

Logiques à injection de courant et à absorption de courant Il est possible de classer les familles logiques selon l'écoulement du courant entre la sortie d'un circuit logique et l'entrée d'un autre. La figure 8-5 a) illustre la logique *à injection de courant*. Quand la sortie de la porte 1 est au niveau HAUT, elle alimente l'entrée de la porte 2 avec un courant I_{IH}, et cette entrée joue essentiellement le rôle d'une résistance raccordée à la masse. Donc la sortie de la porte 1 joue le rôle d'une *alimentation* de courant pour l'entrée de la porte 2.

Par ailleurs, la logique à *absorption de courant* est schématisée à la figure 8-5 b). Sur celle-ci, les circuits d'entrée de la porte 2 sont associés à une résistance reliée à $+V_{CC}$, la borne positive d'une alimentation. Quand la sortie de la porte 1 est au niveau BAS, le courant va dans le sens indiqué, soit de l'entrée de la porte 2 à travers la résistance de sortie de la porte 1 jusqu'à la masse. Autrement dit, quand un circuit est à l'état BAS et qu'il pilote une entrée de la porte 2, il doit être en mesure *d'absorber* un courant, I_{IL}, fourni par l'entrée.

La distinction entre circuit logique à injection de courant et circuit logique à absorption de courant est importante, comme vous allez vous en apercevoir à mesure que nous étudierons les différentes familles logiques.

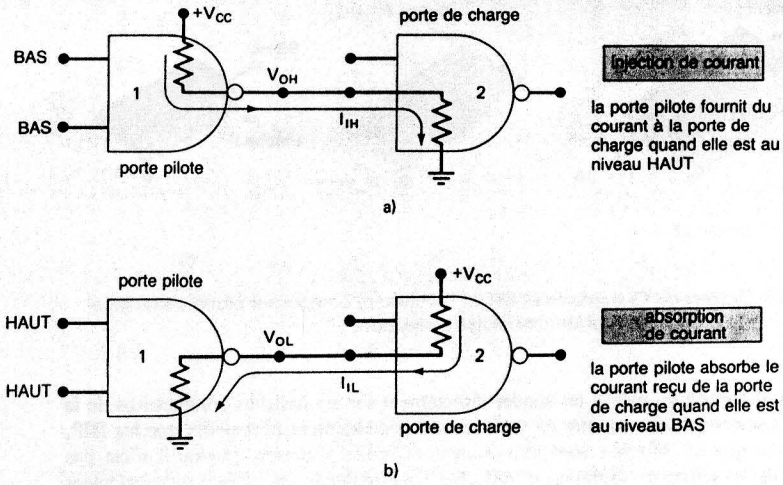


FIGURE 8-5 Comparaison du mode de fonctionnement de la logique à injection de courant avec celui de la logique à absorption de courant.

Degré d'intégration des CI Voici un tableau qui résume les cinq degrés d'intégration des CI.

Degré d'intégration	Nombre de portes
Intégration à petite échelle (SSI)	Moins de 12
Intégration à moyenne échelle (MSI)	12 à 99
Intégration à grande échelle (LSI)	100 à 9999
Intégration à très grande échelle (VLSI)	10 000 à 99 999
Intégration à ultragrande échelle (ULSI)	100 000 et plus

Boîtiers de circuits intégrés Les circuits intégrés sont encapsulés dans plusieurs types de boîtiers. Ces boîtiers se distinguent par le nombre de circuits intégrés sur la puce, le nombre de broches externes, les conditions ambiantes de fonctionnement et la méthode de montage sur la plaquette de circuits. La figure 8-6 nous montre les boîtiers les plus courants.

Le boîtier illustré à la figure 8-6 a) est le composant DIP (double rangée en ligne) introduit au chapitre 4, plus précisément il s'agit d'un élément de 24 broches. Remarquez l'encoche à une des extrémités. Quand vous placez cette encoche à gauche, la broche 1 est celle immédiatement sous l'encoche. Certains composants DIP ont des encoches aux deux extrémités et un petit point en relief sur ces boîtiers indique l'emplacement de la broche 1.

La figure 8-6 b) illustre un *boîtiers plat en céramique*. Il s'agit d'une enveloppe plate et hermétique formée d'une base céramique non conductrice qui met les circuits entièrement à l'abri de l'humidité. Ces boîtiers se retrouvent souvent dans les applications militaires qui doivent fonctionner dans des conditions ambiantes très rigoureuses.

La figure 8-6 c) représente le type de boîtier le plus récent: il s'agit d'un composant *monté en surface*, analogue au DIP à la seule différence des broches qui sont courbées à

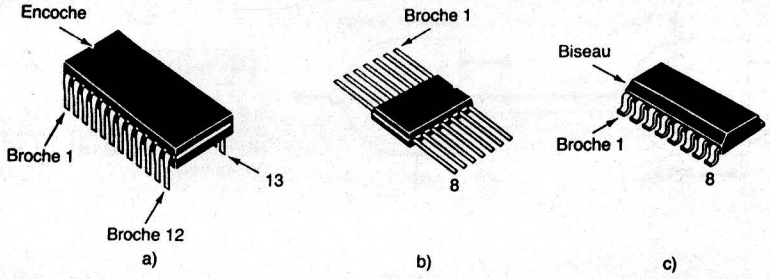


FIGURE 8-6 Boîtiers de CI courants: a) DIP 24 broches; b) composant plat à céramique 14 broches; c) boîtier à montage en surface.

angle droit afin que l'on puisse les souder directement sur les pastilles conductrices de la plaquette. Les composants montés en surface sont généralement plus petits que les DIP, surtout parce que les broches sont plus minces et moins robustes, puisqu'il n'est pas nécessaire de les enfoncer dans un raccord. Les CI à montage en surface ont également l'avantage d'être plus faciles à manipuler par les placeuses automatiques durant la fabrication des plaquettes de circuits. Observez l'emplacement de la broche 1 par rapport au côté bisauté du boîtier.

QUESTIONS DE RÉVISION

1. Définissez chacun des paramètres suivants: V_{OH}, V_{IL}, I_{OL}, I_{IH}, t_{PLH}, t_{PHL}, I_{CCL}, I_{CCH}.
2. *Vrai ou faux:* Si la sortance d'un circuit logique est de 5, cela veut dire que ce circuit a 5 bornes de sortie.
3. Définissez V_{NL} et V_{NH}.
4. *Vrai ou faux:* Une famille logique ayant t_{pd}(moy) = 12 ns et P_D(moy) = 15 mW a un produit vitesse-consommation supérieur à celui d'une famille ayant 8 ns et 30 mW.
5. Quelle est la différence entre l'injection de courant et l'absorption de courant?
6. Quel type de boîtier ne possède pas une encoche de repérage de la broche 1?
7. Quel boîtier a été conçu pour résister à l'humidité?

8-2 FAMILLE LOGIQUE TTL

Présentement, la famille TTL (logique-transistor-transistor) demeure celle que l'on retrouve le plus souvent dans les applications exigeant des dispositifs SSI et MSI. Le circuit logique TTL de base est une porte NON-ET. Ce circuit, schématisé à la figure 8-7 a), manifeste plusieurs caractéristiques particulières. D'abord, remarquez que le transistor Q_1 possède deux émetteurs; par conséquent, il y a deux jonctions émetteur-base qui peuvent mettre Q_1 dans l'état CONDUCTEUR. Un transistor d'entrée comme celui-ci peut avoir jusqu'à ***huit émetteurs*** pour constituer une porte NON-ET à 8 entrées.

Notez ensuite que les deux transistors du côté sortie de ce circuit, Q_3 et Q_4, sont montés en **totem**. Comme vous l'apprendrez bientôt, en fonctionnement normal, c'est le transistor Q_3 ou Q_4 qui conduit suivant la valeur de l'état logique de la sortie.

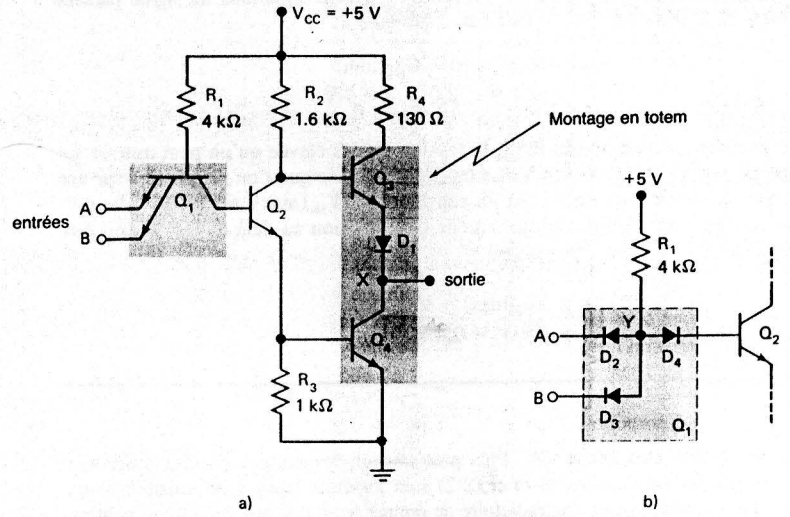


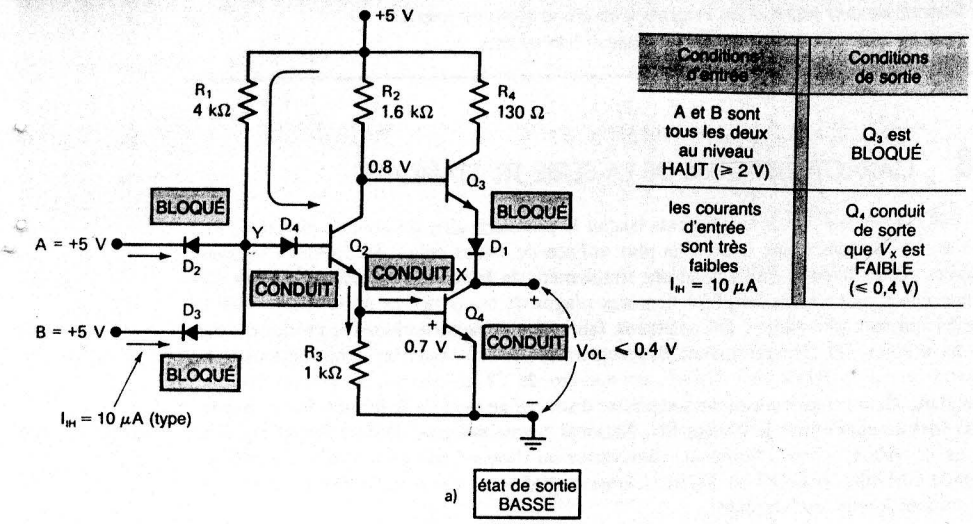
FIGURE 8-7 a) Porte NON-ET TTL de base; b) circuit équivalent à diodes de Q_1.

Fonctionnement du circuit-état BAS Même si à première vue ce circuit semble complexe, il est possible de simplifier l'analyse de son fonctionnement en remplaçant le transistor **multi-émetteur** Q_1 par son circuit équivalent à diodes, montré à la figure 8-7 b). Les deux diodes D_2 et D_3 correspondent aux deux jonctions E-B de Q_1, et D_4 à la jonction collecteur-base du même transistor. Dans l'analyse qui suit, c'est ce circuit équivalent que nous utilisons pour représenter Q_1.

D'abord, étudions le cas où la sortie est BASSE. La figure 8-8 a) schématise cette situation, et les entrées A et B sont toutes les deux à +5 V. Cette tension de +5 V appliquée aux cathodes de D_2 et D_3 BLOQUE les diodes, qui ne laissent passer presque aucun courant. La tension d'alimentation de +5 V force un courant à traverser R_1 et D_4 et la base de Q_2, ce qui place ce dernier dans l'état CONDUCTEUR. Le courant qui traverse l'émetteur de Q_2 se rend jusqu'à la base de Q_4 et place ce dernier en CONDUCTION. Au même moment, le courant de collecteur de Q_2 donne lieu à une différence de potentiel aux bornes de R_2 qui amène la tension du collecteur de Q_2 à une valeur trop basse pour mettre Q_3 en CONDUCTION.

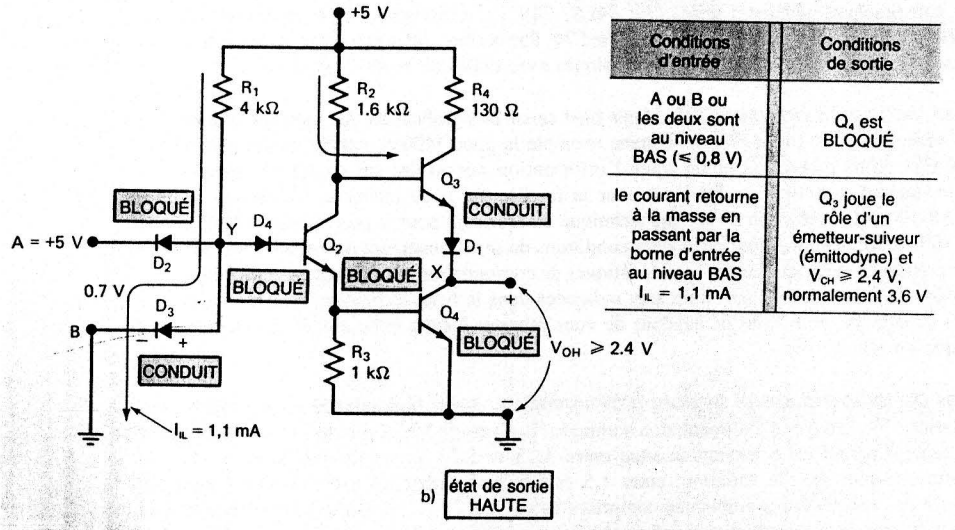
La tension du collecteur Q_2 est, comme on le voit, environ 0,8 V. Ceci parce que l'émetteur Q_2 est approximativement à 0,7 V par rapport à la masse en raison de la tension de polarisation E-B de Q_4 et que le collecteur de Q_2 est approximativement à 0,1 V par rapport à son émetteur à cause de V_{CE}(saturé). Une tension de 0,8 V à la base de Q_3 ne suffit pas à polariser en direct la jonction E-B de Q_3 et la diode D_1. En fait, le rôle de D_1 est de garder Q_3 BLOQUÉ quand prévaut cette situation.

Quand Q_4 CONDUIT, la borne de sortie X a une tension très faible, car la résistance de Q_4 lorsqu'il conduit est très faible (1 à 25 Ω). En réalité, la tension de sortie, V_{OL}, est fonction de l'intensité du courant de collecteur de Q_4. Étant donné que Q_3 est BLOQUÉ, il n'y a pas de courant délivré par la borne +5 V qui traverse R_4. Comme nous le verrons, le courant de collecteur de Q_4 provient des entrées TTL que l'on a raccordées à la borne X.



Conditions d'entrée	Conditions de sortie
A et B sont tous les deux au niveau HAUT (≥ 2 V)	Q_3 est BLOQUÉ
les courants d'entrée sont très faibles I_{IH} = 10 μA	Q_4 conduit de sorte que V_X est FAIBLE (≤ 0,4 V)

a) état de sortie BASSE



Conditions d'entrée	Conditions de sortie
A ou B ou les deux sont au niveau BAS (≤ 0,8 V)	Q_4 est BLOQUÉ
le courant retourne à la masse en passant par la borne d'entrée au niveau BAS I_{IL} = 1,1 mA	Q_3 joue le rôle d'un émetteur-suiveur (émittodyne) et $V_{OH} \geq 2,4$ V, normalement 3,6 V

b) état de sortie HAUTE

FIGURE 8-8 Porte NON-ET TTL représentée pour ses deux valeurs de sortie possibles.

Il importe de remarquer que les entrées au niveau HAUT appliquées à A et B ne fournissent que le courant de fuite des diodes qui est très petit; habituellement I_{IH} vaut environ 10 μA à la température de la pièce.

Fonctionnement du circuit-état HAUT La situation qu'on observe quand la sortie du circuit est au niveau HAUT est schématisée à la figure 8-8 b). Cette situation se produit quand l'une des deux entrées est au niveau BAS ou que les deux entrées sont au niveau

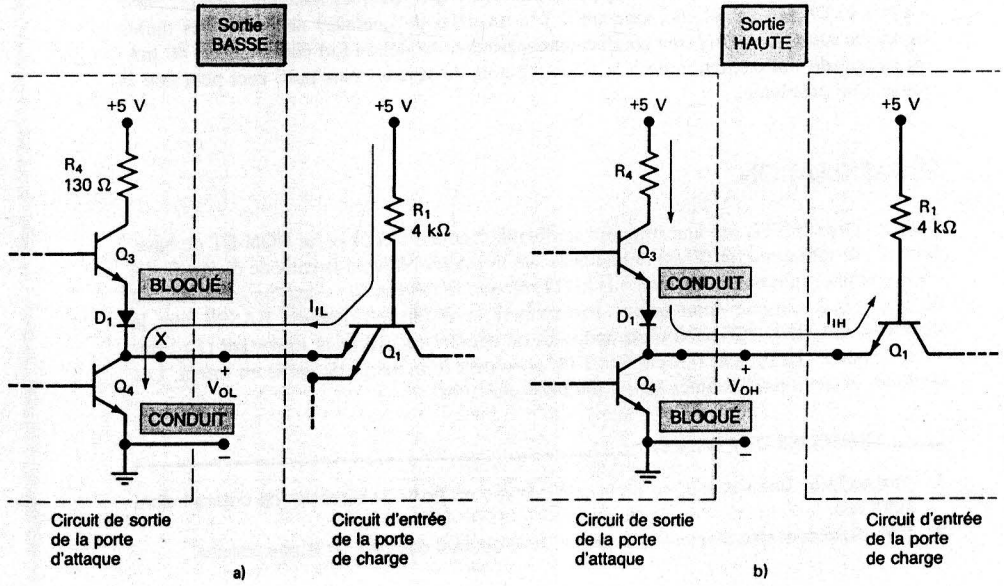
BAS. Dans le cas présent, c'est l'entrée B qui est connectée à la masse. Ceci a pour effet de polariser en direct D_3, de sorte qu'il y a un courant qui part de la borne de l'alimentation +5 V, qui traverse R_1 et D_3 jusqu'à la borne B et la masse. La différence de potentiel aux bornes de D_3 garde le point Y à environ 0,7 V. Cette tension n'est pas suffisante pour polariser en direct D_4 ni la jonction E-B de Q_2 qui, de ce fait, reste bloquée.

Étant donné que Q_2 est BLOQUE, Q_4 n'a pas de courant de base et il passe lui aussi à l'état BLOQUÉ. Comme Q_2 n'a pas de courant de collecteur, la tension à la base de Q_3 est suffisamment élevée pour polariser en direct ce transistor et D_1, et pour mettre ce transistor en conduction. En réalité, Q_3 joue le rôle d'un émetteur-suiveur puisque la tension en X est essentiellement la même que sur l'émetteur de Q_3. Quand aucune charge n'est raccordée entre le point X et la masse, V_{OH} se situe autour de 3,4 à 3,8 V, en raison des deux différences de potentiel de 0,7 V des diodes (E-B de Q_3 et D_1) dont on retranche la somme de la tension de 5 V appliquée à la base de Q_3. Toutefois, cette tension diminue lorsqu'il y a une charge, parce que la charge prélève du courant de l'émetteur de Q_3, ce qui provoque un courant de base à travers R_2 et a pour effet, en dernier lieu, d'augmenter la différence de potentiel aux bornes de R_2.

Il est important de remarquer qu'un courant assez intense retourne à la masse en passant par la borne B. Ce courant, I_{IL}, est généralement de l'ordre de 1,1 mA. L'entrée B au niveau BAS joue le rôle d'un *point où se perd* ce courant dans la masse.

Absorption de courant Une sortie TTL agit comme un élément d'absorption du courant quand elle est au niveau BAS, puisqu'elle *reçoit* du courant de l'entrée de la porte qu'elle pilote. La figure 8-9 montre une porte TTL pilotant l'entrée d'une autre porte (la charge) pour les deux niveaux logiques possibles. Quand la sortie est au niveau BAS, cas illustré

FIGURE 8-9 a) Quand la sortie TTL est à l'état BAS, Q_4 se comporte comme un absorbeur de courant qui tire son courant de l'entrée; b) quand elle est à l'état HAUT, Q_3 agit comme un fournisseur de courant à la porte de charge.



par la figure 8-9 a), le transistor Q_4 de la porte d'attaque conduit et est, en pratique, un « court-circuit » entre le point X et la masse. Cette tension BASSE sur X polarise en direct la jonction émetteur-base de Q_1 et le courant revient à travers Q_4, comme l'indique le sens de la flèche. On constate donc que ce transistor a un effet d'absorption du courant, et qu'il prélève son courant du courant (I_{IL}) de la charge. Nous désignerons souvent Q_4 comme le **transistor d'absorption de courant**. Parfois, nous le qualifierons de **transistor de rappel à 0** parce qu'il ramène la sortie à son niveau BAS.

Injection de courant Une sortie TTL agit comme une source de courant quand elle se trouve à son niveau HAUT. C'est ce qu'illustre la figure 8-9 b), où le transistor Q_3 fournit le courant d'entrée I_{IH} nécessaire au fonctionnement du transistor Q_1 de la porte de charge. Comme nous l'avons déjà dit, il s'agit d'un faible courant de fuite qui le polarise en inverse (généralement 10 μA). Souvent, nous nous référerons à Q_3 en parlant du **transistor d'injection de courant** ou **transistor de rappel à la source**.

Circuit de sortie en totem Il faut faire remarquer plusieurs points concernant le montage en totem, puisque le choix d'un tel montage n'est pas particulièrement évident. On obtiendrait la même réponse logique si on enlevait Q_3 et D_1 et que l'on raccordait la borne inférieure de R_4 au collecteur de Q_4. Dans un tel cas toutefois, Q_4 devrait supporter un courant de saturation particulièrement élevé (5 V/130 $\Omega \approx$ 40 mA). Avec Q_3 dans le circuit, il n'y a pas de courant qui traverse R_4 lorsque l'état de sortie est BAS. Il est important qu'il en soit ainsi afin que le circuit ne consomme pas trop.

Un autre avantage de ce montage s'observe quand l'état de sortie est HAUT. Dans ce cas-ci, Q_3 agit comme un émetteur-suiveur, montage dont on sait que l'impédance de sortie est basse (habituellement 10 Ω). Grâce à cette impédance de sortie basse, la constante de temps nécessaire pour charger toute charge capacitive en sortie est courte. Cette action (appelée souvent **rappel à la source actif**) donne lieu à des formes d'ondes de sortie TTL dont le temps de montée est très rapide.

Un inconvénient du montage avec sortie en totem s'observe lors de la transition de BAS à HAUT. Malheureusement, Q_4 passe à l'état BLOQUÉ plus lentement que Q_3 passe à l'état CONDUCTEUR, de sorte qu'il y a un délai de quelques nanosecondes durant lequel les deux transistors sont conducteurs et pendant lequel un fort courant (30 à 40 mA) est soutiré de l'alimentation de 5 V. Comme nous le verrons plus tard, ceci peut être la cause d'un problème.

RÉCAPITULATION

Tous les circuits TTL ont une structure semblable à celle de cette porte NON-ET de base. L'entrée de tout circuit TTL est l'émetteur d'un transistor NPN ou la cathode d'une diode; c'est pourquoi une tension d'entrée HAUTE bloque le dispositif — seul un petit courant de fuite (I_{IH}) traverse cette borne. Inversement, une tension d'entrée BASSE met le dispositif en CONDUCTION; c'est pourquoi un courant relativement important (I_{IL}) traverse la borne. La plupart des circuits TTL possèdent le montage de sortie en totem. Les quelques exceptions existantes seront étudiées plus loin.

QUESTIONS DE RÉVISION

1. Vrai ou faux: Une sortie TTL se comporte comme un point d'absorption de courant dans l'état BAS.
2. Dites dans quel état d'entrée d'un circuit TTL le courant d'entrée est le plus intense.
3. Donnez les avantages et les inconvénients d'une sortie en totem.
4. Quel transistor TTL est le transistor de rappel à la source.

8-3 CARACTÉRISTIQUES DE LA SÉRIE TTL STANDARD

En 1964, la Société Texas Instruments lançait la première série de circuits intégrés TTL, la série 54/74, sans doute la série la plus utilisée de toutes celles des familles logiques intégrées. Dans la suite, nous parlerons simplement de la série 74, puisque la seule différence entre les versions 54 et 74 tient aux plages de températures et de tensions de la série 54 qui sont plus larges. De nombreux fabricants de semiconducteurs produisent des circuits intégrés TTL. Heureusement, ils acceptent de se conformer à la même numérotation, de sorte que d'un fabricant à l'autre, un numéro de CI correspond au même élément. Cependant, chaque fabricant ajoute son préfixe distinctif au numéro de la puce. Par exemple, Texas Instruments utilise le préfixe SN, National Semiconductor, DM et Signetics, simplement S. Ainsi, selon le fabricant, vous verrez qu'il existe une puce à quatre portes NI désignée DM7402, SN7402 ou S7402. L'important dans ce cas c'est le numéro 7402 qui est commun à tous les fabricants.

Vous vous souvenez qu'au chapitre 4 nous vous avons appris que la famille TTL était constituée de plusieurs séries (74, 74LS, 74S...). Commençons par examiner les caractéristiques électriques de la série standard 74. Par la suite, nous présenterons les autres séries TTL et comparerons leurs caractéristiques avec celles de la série standard.

Fiches techniques des fabricants Pour bien saisir la signification des caractéristiques de la série 74, nous allons prendre comme exemple la puce 7400 contenant quatre portes NON-ET. Nous pouvons trouver toute l'information nécessaire sur un CI en consultant le manuel technique du fabricant pour la famille qui nous intéresse. D'ailleurs, la figure 8-10 est la traduction de la fiche technique du fabricant pour la porte intégrée NON-ET 5400/7400; cette dernière précise les conditions de fonctionnement recommandées, les caractéristiques électriques et les caractéristiques de commutation. La plupart des grandeurs discutées dans les sections suivantes sont indiquées dans la fiche technique. Pour chacune de ces grandeurs, nous vous demandons de vous reporter à cette fiche afin d'identifier la rubrique où elle se trouve.

Plages de températures et tensions d'alimentation Aussi bien pour la série 74 que pour la série 54, la tension d'alimentation nominale (V_{CC}) est de 5 V. La série 74 fonctionne correctement quand cette tension se situe entre 4,75 et 5,25 V, tandis que la série 54 supporte une plus grande variation: entre 4,5 et 5,5 V. La série 74 a été conçue pour résister à des températures ambiantes comprises entre 0 et 79 °C. Par contre, la série 54 conserve toutes ses caractéristiques même dans des conditions plus rigoureuses: sa plage de températures admissibles va de -55 à $+125$ °C. À cause de sa plus grande tolérance aux écarts de températures et de tension, la série 54 est plus coûteuse. On ne l'utilise que dans des applications où la fiabilité est primordiale même dans des conditions ambiantes extrêmes. Par exemple, dans les équipements militaires ou dans les engins spatiaux.

Niveaux de tension Le tableau 8-1 nous fait connaître les niveaux de tension d'entrée et de sortie que l'on doit respecter avec la série standard 74. Les valeurs minimales et maximales indiquées correspondent au cas pessimiste des conditions d'alimentation, de température et de charge. L'examen de ce tableau montre que la sortie maximale garantissant

un niveau logique 0 est V_{OL} = 0,4 V, ce qui est 400 mV de moins que la tension à l'entrée nécessaire pour avoir un 0 logique, soit V_{IL} = 0,8 V. On voit donc que la marge de sensibilité aux bruits CC pour l'état BAS est 400 mV. Donc

$$V_{NL} = V_{IL}(\text{max}) - V_{OL}(\text{max}) = 0{,}8\text{ V} - 0{,}4\text{ V} = 0{,}4\text{ V} = 400\text{ mV}$$

TABLEAU 8-1 NIVEAUX DE TENSION DE LA SÉRIE 74 STANDARD

	Minimal	Typique	Maximal
V_{OL}	—	0,2	0,4
V_{OH}	2,4	3,6	—
V_{IL}	—	—	0,8
V_{IH}	2,0	—	—

De même, la sortie minimale qui garantit un 1 logique, V_{OH}, est 2,4 V, ce qui est 400 mV de plus que la tension d'entrée nécessaire pour avoir un 1 logique, soit V_{IH} = 2,0 V. Donc la marge de sensibilité aux bruits CC pour l'état HAUT est 400 mV.

$$V_{NH} = V_{OH}(\text{min}) - V_{IH}(\text{min}) = 2{,}4\text{ V} - 2{,}0\text{ V} = 0{,}4\text{ V} = 400\text{ mV}$$

Et les marges de sensibilité CC de la série 74 *garanties pour le cas pessimiste* sont toutes les deux 400 mV. En pratique, les marges *typiques* sont supérieures à cela (V_{NL} = 1 V et V_{NH} = 1,6 V).

Tensions maximales nominales Les tensions indiquées au tableau 8-1 ne donnent pas les valeurs maximales nominales au-delà desquelles la durée de vie utile d'un CI peut être écourtée. Les tensions appliquées à toute entrée d'un CI de la série 74 standard ne doivent jamais dépasser +5,5 V. Une tension supérieure à cela, appliquée à un émetteur d'entrée, peut provoquer le claquage de la jonction E-B de Q_1 polarisée en inverse.

Une limite est également imposée à la tension maximale *négative* que l'on peut appliquer à une entrée TTL. Cette limite, −0,5 V, apparaît du fait que la plupart des circuits TTL font usage de diodes de protection montées en parallèle pour chaque entrée. Ces diodes n'ont pas été prises en considération lors de notre analyse précédente, puisqu'elles ne modifient en rien le fonctionnement normal du circuit. Elles sont connectées entre chaque entrée et la masse afin de limiter les excursions de tension négative à l'entrée, qui s'observent souvent quand les signaux logiques ont des oscillations transitoires excessives. Il ne faut donc pas appliquer de tension négative plus basse que −0,5 V à une entrée parce que les diodes de protection commencent à conduire et à absorber un courant assez intense.

Consommation Une porte NON-ET standard draine une puissance moyenne de 10 mW. On déduit cette valeur de I_{CCH} = 4 mA et I_{CCL} = 12 mA, pour une intensité moyenne I_{CC}(moy) = 8 mA et une consommation P_D(moy) = 8 mA × 5 V = 40 mW. Comme c'est la consommation pour les quatre portes de la puce, la consommation d'une porte est de 10 mW.

Conditions de fonctionnement recommandées

		SN5400			SN7400			UNITÉ
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Tension d'alimentation	4.5	5	5.5	4.75	5	5.25	V
V_{IH}	Tension d'entrée niveau haut	2			2			V
V_{IL}	Tension d'entrée niveau bas			0.8			0.8	V
I_{OH}	Courant de sortie niveau haut			−0.4			−0.4	mA
I_{OL}	Courant de sortie niveau bas			16			16	mA
T_A	Température de fonctionnement dans le vide	−55		125	0		70	°C

Caractéristiques électriques pour la plage de températures de fonctionnement recommandée dans le vide (sauf indication contraire)

PARAMÈTRE	CONDITIONS DE TEST†			SN5400			SN7400			UNITÉ
				MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IK}	V_{CC} = MIN,	I_I = −12 mA				−1.5			−1.5	V
V_{OH}	V_{CC} = MIN,	V_{IL} = 0.8 V,	I_{OH} = −0.4 mA	2.4	3.4		2.4	3.4		V
V_{OL}	V_{CC} = MIN,	V_{IH} = 2 V,	I_{OL} = 16 mA		0.2	0.4		0.2	0.4	V
I_I	V_{CC} = MAX,	V_I = 5.5 V				1			1	mA
I_{IH}	V_{CC} = MAX,	V_I = 2.4 V				40			40	µA
I_{IL}	V_{CC} = MAX,	V = 0,4 V				−1.6			−1.6	mA
I_{OS}§	V_{CC} = MAX			−20		−55	−18		−55	mA
I_{CCH}	V_{CC} = MAX,	V_I = 0 V			4	8		4	8	mA
I_{CCL}	V_{CC} = MAX,	V_I = 4,5 V			12	22		12	22	mA

† Pour les conditions désignées MIN ou MAX, utilisez la valeur sous la rubrique Conditions de fonctionnement recommandées.
‡ Toutes les valeurs types sont pour V_{CC} = 5 V et T_A = 25 °C.
§ Il ne doit pas y avoir plus d'une sortie de court-circuitée à la fois.

Caractéristiques de commutation V_{CC} = 5 V, T_A = 25 °C (voir note 2)

PARAMÈTRE	DE (ENTRÉE)	À (SORTIE)	CONDITIONS DE TEST		MIN	TYP	MAX	UNITÉ
t_{PLH}	A ou B	Y	R_L = 400 Ω,	C_L = 15 pF		11	22	ns
t_{PHL}						7	15	ns

Note 2: Pour les circuits de charge et les formes d'ondes des tensions, reportez-vous à la rubrique Informations générales.

FIGURE 8-10 Fiche technique de la porte intégrée NON-ET 7400 (Gracieuseté de Texas Instruments).

Sortance La sortie d'une TTL standard peut normalement piloter 10 entrées TTL standard. Certaines fiches techniques désignent explicitement la sortance du dispositif, contrairement à la fiche de la figure 8-10. Plus loin, nous verrons que la sortance peut être déterminée à partir des courants d'entrée et de sortie indiqués dans cette figure.

Le tableau 8-2 résume les caractéristiques d'une porte NON-ET de la série standard 74.

TABLEAU 8-2 CARACTÉRISTIQUES DE LA SÉRIE STANDARD 74

Marges de sensibilité aux bruits (cas pessimiste)	$V_{NL} = V_{NH}$ = 400 mV
Consommation moyenne (porte de base)	P_D = 10 mW
Retard de propagation moyen (porte de base)	9 ns
Sortance type	10

Retards de propagation La porte NON-ET TTL de base présente des retards de propagation types de t_{PLH} = 11 ns et t_{PHL} = 7 ns, ce qui donne un retard de propagation *moyen* de 9 ns. On résume au tableau 8.2 les caractéristiques de la porte NON-ET de la série 7400 standard.

EXEMPLE 8-2

Reportez-vous à la fiche technique de la puce 7400 ayant quatre portes NON-ET à deux entrées, reproduite à la figure 8-10. Déterminez la consommation moyenne maximale et le retard de propagation moyen maximal d'une *seule* porte.

Solution Allez à la rubrique caractéristiques électriques pour trouver les valeurs maximales I_{CCH} et I_{CCL}, soit respectivement 8 mA et 22 mA. I_{CC} moyen est donc (8 + 22)/2 = 15 mA. La consommation moyenne est obtenue en multipliant cette dernière valeur par V_{CC}. La fiche technique indique que ces valeurs de I_{CC} correspondent à V_{CC} maximal (5,25 pour la série 74). Il vient donc

$$P_D(\text{moy}) = 15 \text{ mA} \times 5{,}25 \text{ V} = 78{,}75 \text{ mW}$$

pour la consommation de *toute* la puce. On trouve la consommation d'une seule porte NON-ET en divisant par quatre:

$$P_D(\text{moy}) = \mathbf{19{,}7 \text{ mW par porte}}$$

Étant donné que la consommation moyenne a été calculée à partir du courant maximal et de la tension maximale, c'est la consommation maximale d'une porte NON-ET 7400 pour le cas « pessimiste ». Les concepteurs font souvent leur analyse pour le cas pessimiste afin d'être certains que leurs circuits fonctionneront bien dans toutes les conditions.

Les retards de propagation maximaux d'une porte NON-ET 7400 sont d'après la fiche:

$$t_{PHL} = 22 \text{ ns}, \qquad t_{PHL} = 15 \text{ ns}$$

donc le retard de propagation moyen est

$$t_{pd}(\text{moy}) = \frac{22 + 15}{2} = \mathbf{18{,}5 \text{ ns}}$$

De nouveau, ce retard de propagation moyen maximal correspond au cas pessimiste.

8-4 AUTRES SÉRIES TTL

Les CI de la série 74 standard offrent un bon compromis entre vitesse et consommation convenant à de nombreuses applications. Les circuits intégrés commercialisés de cette série contiennent un large éventail de portes, de bascules, de monostables, dans la gamme des produits d'intégration à petite échelle (SSI), et de registres à décalage, de compteurs, de décodeurs, de mémoires et de circuits arithmétiques dans la gamme des produits d'intégration à moyenne échelle (MSI).

Après l'avènement de la série 74 standard, d'autres séries TTL ont été mises au point. Dans ces autres séries, les caractéristiques de vitesse et de consommation sont très différentes. Nous décrivons ces séries dans les paragraphes suivants. Remarquez que lorsqu'on utilise le sigle « TTL », on se réfère généralement à la série 74 standard.

TTL faible consommation, série 74 Les circuits TTL faible consommation, désignés sous le nom série 74L, sont essentiellement les mêmes circuits que ceux de la série standard sauf pour ce qui est des résistances qui sont toutes *plus élevées*. En utilisant des résistances plus élevées, on réduit la consommation mais au détriment des retards de propagation qui sont désormais plus longs. Une porte NON-ET caractéristique de cette série consomme en moyenne 1 mW mais a un retard de propagation moyen de 33 ns.

La série 74L est idéale pour les applications dans lesquelles la consommation est un élément plus important que la vitesse. Les circuits à pile et à basse fréquence comme les calculatrices conviennent bien à cette série TTL. Cette série a la plus faible consommation d'énergie de toutes les séries TTL.

Pour l'essentiel, la série 74L est désormais périmée depuis l'arrivée des séries 74LS, 74ALS et CMOS, qui déclassent la série 74L autant pour ce qui concerne la consommation que la vitesse d'exécution. Pour cette raison, on recommande de ne plus utiliser cette série dans les nouveaux produits.

TTL rapide, série 74H La série 74H est une série TTL rapide. Les circuits de base de cette série sont essentiellement les mêmes que ceux de la série 74 standard, à l'exception des résistances qui sont toutes *plus petites* et du transistor émetteur-suiveur Q_3 remplacé par un montage Darlington. Ces changements donnent une vitesse de commutation plus rapide et un retard de propagation moyen de 6 ns. Toutefois, l'augmentation de la vitesse se fait aux dépens d'une consommation accrue. La porte NON-ET de base dans cette série a une consommation P_D de 23 mW. À toutes fins pratiques, la série 74H est désuète depuis l'arrivée de la série TTL Schottky décrite dans les paragraphes suivants, et nous ne la recommandons pas dans les nouveaux projets.

TTL Schottky, série 74S Les séries 74, 74L et 74H fonctionnent toutes avec des transistors qui sont à saturation lorsqu'ils conduisent. Le fonctionnement à saturation provoque un retard de stockage, T_S, quand les transistors passent de l'état BLOQUÉ à l'état CONDUCTEUR, ce qui limite la vitesse de commutation du circuit.

La série 74S réduit ce retard de stockage en interdisant au transistor de devenir complètement à saturation. C'est une diode à barrière Schottky (DBS) raccordée entre la base et le collecteur de chaque transistor qui accomplit cela. Voir figure 8-11 a). La DBS a une tension de polarisation directe de seulement 0,25 V. Aussi, quand la jonction C-B devient polarisée en direct au début de la saturation, la DBS conduit et dévie une partie du courant d'entrée de la base. Ceci réduit le courant de base excédentaire et diminue le retard de stockage lorsqu'on met le transistor à l'état BLOQUÉ.

Comme on peut le voir à la figure 8-11 a), on utilise un symbole spécial pour désigner la combinaison transistor/DBS. D'ailleurs à la figure 8-11 b), on retrouve ce symbole pour tous les transistors du circuit de la porte NON-ET 74S00, sauf Q_4. Cette porte NON-ET a un retard de propagation moyen d'à peine 3 ns, ce qui est deux fois plus rapide que la porte 74H. Remarquez la présence des diodes de protection montées en parallèle, D_1 et D_2, qui limitent les tensions d'entrée négatives.

Les circuits de la série 74S sont également associés à des résistances plus petites afin d'améliorer les temps de commutation. Ceci augmente la consommation moyenne du circuit qui atteint environ 20 mW, presque la même chose que les 74H. Les circuits 74S utilisent également un montage Darlington (Q_3 et Q_4) pour obtenir en sortie un temps de montée plus rapide au moment de la commutation état CONDUCTEUR à état BLOQUÉ.

Donc, les 74S sont deux fois plus rapides que les 74H pour à peu près la même consommation. C'est pourquoi les 74H ont été supplantés.

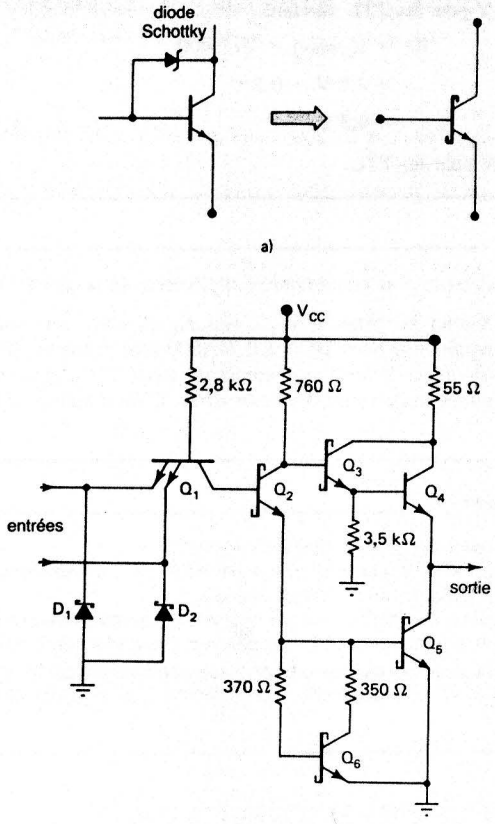


FIGURE 8-11 a) Transistor insaturable Schottky; b) porte NON-ET de base dans la série 74SL. (Gracieuseté de Fairchild, filiale Schlumberger).

TTL Schottky faible consommation, série 74LS (LS-TTL) Il s'agit ici d'une variante moins gourmande et également moins rapide de la série 74S. Cette série utilise les mêmes transistors Schottky insaturables de la série 74S mais des résistances plus élevées (voir figure 8-12). Aussi, ce circuit consomme-t-il moins, mais au prix de temps de commutation plus longs.

Une porte NON-ET de la série 74LS a comme retard de propagation moyen 9,5 ns et comme consommation moyenne, 2 mW. Étant donné qu'elle a presque la même vitesse de commutation que la série TTL standard mais qu'elle consomme moins, la série 74LS tend à remplacer graduellement la série 74 dans les applications où le fonctionnement est relativement rapide mais où la consommation doit être minimale. Donc les 74LS sont devenus les circuits « ordinaires » de la famille TTL et on les retrouve maintenant dans presque toutes les nouvelles conceptions TTL dans lesquelles la vitesse n'a pas à être maximale. Sa position dominante s'amenuise de plus en plus face à l'émergence de la série 74ALS.

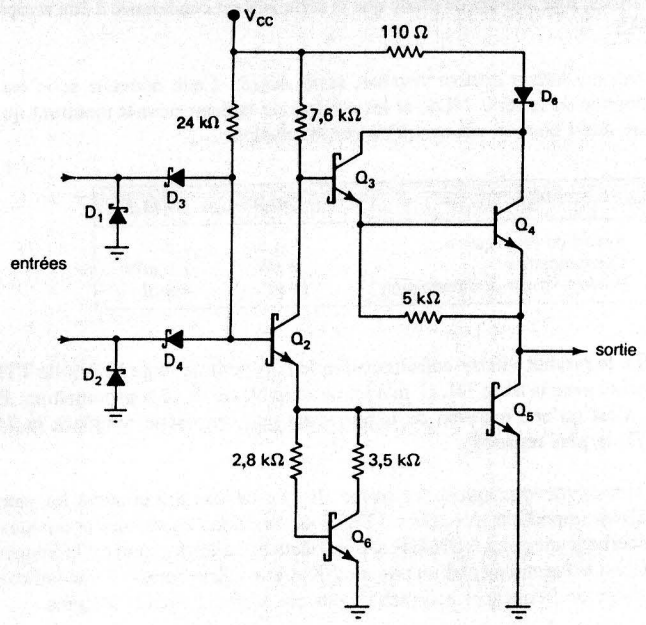


FIGURE 8-12 Porte NON-ET 74LS00 (Gracieuseté de Fairchild, filiale Schlumberger).

Notez que la porte NON-ET 74LS de la figure 8-12 n'a pas un transistor d'entrée à plusieurs émetteurs. On y retrouve plutôt des diodes d'entrée (D_3 et D_4); malgré cela le fonctionnement de base de ce circuit est identique à celui avec transistor à émetteur multiple.

TTL Schottky avancé, série 74ALS (AS-TTL) De récentes percées dans la conception des circuits intégrés ont conduit à la mise au point de deux nouvelles séries TTL améliorées: la série Schottky avancé (74AS) et la série Schottky avancé faible consommation (74ALS). La série 74AS constitue un progrès considérable par rapport à la série 74S, puisqu'elle fonctionne plus rapidement en consommant moins. Voici un tableau qui compare ces deux séries:

	74S	74AS
Retard de propagation	3 ns	1.7 ns
Consommation	20 mW	8 mW
Produit vitesse-consommation	60 pJ	13.6 pJ

La comparaison ne laisse aucun doute sur la supériorité de la série 74AS. D'ailleurs, celle-ci regroupe les composants TTL les plus rapides qui soient, et cela pour un produit vitesse-consommation inférieur à celui de la série 74S. La série 74AS présente en outre d'autres avantages, notamment des besoins en courants d'entrée plus faibles (I_{IL}, I_{IH}), ce qui se traduit par une sortance accrue.

Compte tenu de tous ces avantages, la 74AS est préférée à la série 74S dans toutes les applications où la rapidité est recherchée. À la vitesse avec laquelle chutent les prix

des dispositifs 74AS, il ne fait pas de doute que la série 74S est condamnée à être remplacée par la série 74AS.

TTL Schottky avancé faible consommation, série 74ALS Cette nouvelle série est une version perfectionnée de la série 74LS, et les chiffres du tableau suivant montrent qu'elle lui est supérieure aussi bien en vitesse qu'en consommation.

	74LS	74ALS
Retard de propagation	9.5 ns	4 ns
Consommation	2 nW	1.2 mW
Produit vitesse-consommation	19 pJ	4.8 pJ

La série 74ALS a le produit vitesse-consommation le plus petit de toute la famille TTL et est presque à égalité avec la série 74L (1 mW) pour la faiblesse de sa consommation. Pour ces raisons, ce n'est qu'une question de temps avant que cette série remplace la 74LS comme série TTL la plus répandue.

Comparaison des caractéristiques des séries TTL Le tableau 8-3 contient les valeurs types des paramètres importants des séries TTL. Pour chacune, les valeurs nominales de performance concernent une porte NON-ET, sauf la fréquence d'horloge qui est la fréquence maximale permettant le basculement d'un bascule J-K. Cette valeur constitue une indication précieuse de la plage de fréquences acceptable pour une série de circuits intégrés.

TABLEAU 8-3 CARACTÉRISTIQUES TYPES DES SÉRIES TTL

	74	74L	74H	74S	74LS	74AS	74ALS
Valeurs nominales de performance							
Retard de propagation (ns)	9	33	6	3	9,5	1,7	4
Consommation (mW)	10	1	23	20	2	8	1,2
Produit vitesse-consommation (pJ)	90	33	138	60	19	13,6	4,8
Fréquence d'horloge max. (MHz)	35	3	50	125	45	200	70
Sortance (même série)	10	20	10	20	20	40	20
Paramètres de tension							
V_{OH}(min)	2,4	2,4	2,4	2,7	2,7	2,5	2,5
V_{OL}(max)	0,4	0,4	0,4	0,5	0,5	0,5	0,4
V_{IH}(min)	2,0	2,0	2,0	2,0	2,0	2,0	2,0
V_{IL}(max)	0,8	0,7	0,8	0,8	0,8	0,8	0,8

EXEMPLE 8-3

En vous servant du tableau 8-3, calculez les marges de sensibilité aux bruits CC d'un CI type 74LS. Comparez vos réponses avec les marges trouvées à la section 8-3 pour la série TTL standard.

Solution

$$V_{NH} = V_{OH}(\text{min}) - V_{IH}(\text{min})$$
$$= 2{,}7\text{ V} - 2{,}0\text{ V}$$
$$= \mathbf{0{,}7\text{ V}}$$

Comparé à V_{NH} = 0,4 V pour les TTL standard.

$$V_{NL} = V_{IL}(\text{max}) - V_{OL}(\text{max})$$
$$= 0{,}8\text{ V} - 0{,}5\text{ V}$$
$$= \mathbf{0{,}3\text{ V}}$$

Comparé à V_{NL} = 0,4 V pour les TTL.

EXEMPLE 8-4

Quelle série TTL peut piloter le plus grand nombre d'éléments de sa série?

Solution La série 74AS dont la sortance est la plus élevée (40). Une telle sortance s'interprète ainsi: par exemple, une porte NON-ET 74AS00 peut entretenir 40 entrées de dispositifs 74AS. Pour déterminer le nombre d'entrées de séries TTL *différentes*, on doit connaître les courants d'entrée et de sortie des deux séries. C'est d'ailleurs le sujet de la prochaine section.

QUESTIONS DE RÉVISION

1. a) Quelle série TTL est idéale pour les hautes fréquences?
   b) Quelle série TTL possède la plus grande marge de sensibilité aux bruits niveau HAUT?
   c) Quelles séries sont à toutes fins pratiques périmées?
   d) Quelle série fait appel à une diode spéciale pour réduire le temps de commutation?
   e) Quelle série convient le mieux pour une application à pile cadencée à 10 MHz?
2. Supposons un coût égal, pourquoi devriez-vous choisir le compteur 74ALS193 plutôt que le 74LS193 ou le 74AS193 pour une application synchronisée par un signal d'horloge de 40 MHz?

8-5 FACTEURS DE CHARGE ET SORTANCE DES TTL

Il est important de bien comprendre les paramètres qui déterminent la sortance, ou facteur de charge, d'une sortie de CI. La figure 8-13 a) illustre une sortie TTL standard dans son état BAS qui pilote plusieurs entrées TTL standard. Le transistor Q_4 conduit et agit comme un absorbeur d'un courant I_{OL} égal à la somme des courants I_{IL} provenant de chacune des entrées. Quand il conduit, la résistance collecteur-émetteur de Q_4 est très faible, mais non nulle, de sorte que ce courant I_{OL} est à l'origine d'une différence de potentiel V_{OL}. Cette tension ne doit pas excéder la limite V_{OL}(max) du circuit en question. Cette dernière valeur impose une valeur maximale à I_{OL}, ce qui, en retour, limite le nombre de charges qui peuvent être attaquées.

Comme exemple de ceci, supposons des CI de la série 74 ayant chacun pour I_{IL} une intensité de courant de 1,6 mA. Au tableau 8-3, on relève V_{OL}(max) = 0,4 V et V_{IL}(max) = 0,8 V. De plus, supposons que Q_4 peut absorber jusqu'à 16 mA avant que sa tension n'atteigne V_{OL}(max) = 0,4 V. On doit conclure de cela qu'il peut piloter 10 charges, soit 16 mA/1,6 mA = 10. Si l'on connecte plus de 10 charges, le courant I_{OL} augmente et porte V_{OL} au-delà de 0,4 V. On se retrouve alors dans une condition indésirable parce que cela réduit la marge de sensibilité aux bruits du CI [rappelez-vous que $V_{NL} = V_{IL}(\text{max}) - V_{OL}(\text{max})$]. En fait, si V_{OL} dépasse V_{IL}(max) = 0,8 V, la sortie est dans l'intervalle indéterminé.

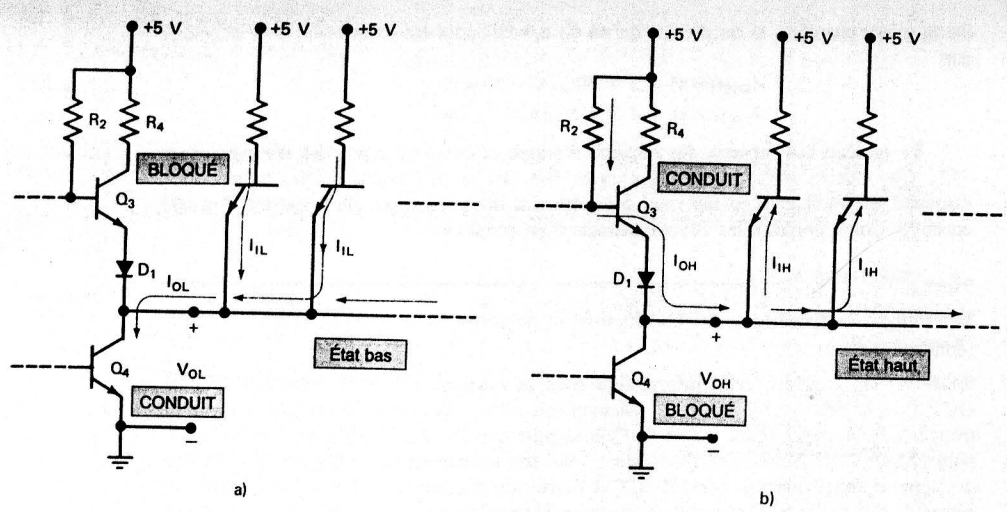


FIGURE 8-13 Courants existants sur la sortie d'un TTL qui pilote plusieurs entrées.

Une situation tout à fait semblable se produit pour l'état HAUT (voyez figure 8-13 b). Dans ce cas-ci, le transistor Q_3 agit comme émetteur suiveur qui fournit (ou injecte) un courant total I_{OH}, soit la somme des courants I_{IH} des différentes entrées TTL. Si l'on raccorde trop de charges, ce courant I_{OH} augmente dans des proportions excessives et les différences de potentiel aux bornes de R_2, la jonction base-émetteur de Q_3 et de D_1 amènent V_{OH} sous $V_{OH}(\text{min})$. Voilà un autre effet indésirable, puisqu'il réduit la marge de sensibilité aux bruits-niveau HAUT; il peut même placer V_{OH} dans l'intervalle indéterminé.

Ces deux discussions montrent qu'une sortie TTL est limitée: par $I_{OL}(\text{max})$, le courant maximal qui peut être absorbé, quand elle est au niveau BAS; ou par $I_{OH}(\text{max})$, le courant maximal fourni, quand elle est au niveau HAUT. Il est essentiel de ne pas dépasser ces courants de sortie limites, si on veut que les niveaux de tension restent à l'intérieur des plages de bon fonctionnement spécifiées.

Détermination de la sortance Pour calculer le nombre d'entrées de séries différentes qu'une sortie de CI peut attaquer, nous devons connaître la capacité d'attaque [soit $I_{OL}(\text{max})$ et $I_{OH}(\text{max})$] de la sortie, de même que les exigences de courant de chacune des entrées (soit I_{IL} et I_{IH}). Ces données figurent toujours dans la fiche technique du fabricant pour le circuit intégré concerné. Voici des exemples décrivant une situation de ce genre.

EXEMPLE 8-5

Combien d'entrées NON-ET 7400 une sortie NON-ET 7400 peut-elle piloter?

Solution Considérons d'abord l'état BAS (illustré à la figure 8-14). Reportez-vous à la fiche technique du 7400, figure 8-10, et relevez les valeurs suivantes:

$$I_{OL}(\text{max}) = 16 \text{ mA}$$
$$I_{IL}(\text{max}) = 1{,}6 \text{ mA}$$

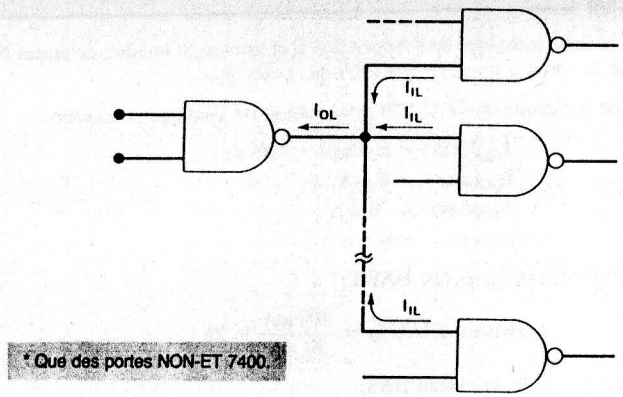


FIGURE 8-14 Exemple 8-5.

Cela signifie qu'une sortie 7400 peut absorber un maximum de 16 mA, et qu'une entrée 7400 fournit un courant maximal de 1,6 mA à la sortie de la porte pilote. On déduit donc que le nombre de portes qu'une sortie peut piloter, quand elle est au niveau BAS, est:

$$\text{Sortance (BAS)} = \frac{I_{OL}(\text{max})}{I_{IL}(\text{max})}$$
$$= \frac{16 \text{ mA}}{1{,}6 \text{ mA}}$$
$$= 10$$

(*Note*: La valeur pour I_{IL} est en réalité $-1{,}6$ mA. Le signe moins sert à indiquer que ce courant *quitte* la borne d'entrée; il n'est pas nécessaire de tenir compte de cela dans cette discussion.) L'état HAUT est analysé de manière similaire. Reportez-vous à la fiche technique et relevez les valeurs pour I_{OH} et I_{IH} en ne tenant pas compte des signes.

$$I_{OH}(\text{max}) = 0{,}4 \text{ mA} = 400 \ \mu\text{A}$$
$$I_{IH}(\text{max}) = 40 \ \mu\text{A}$$

On en conclut que le nombre d'entrées qu'un état HAUT peut piloter est:

$$\text{Sortance (HAUT)} = \frac{I_{OH}(\text{max})}{I_{IH}(\text{max})}$$
$$= \frac{400 \ \mu\text{A}}{40 \ \mu\text{A}}$$
$$= 10$$

La sortance, dans ce cas-ci, est la même pour les deux états de tension. Donc, une porte NON-ET 7400 peut piloter jusqu'à 10 portes NON-ET 7400. Quand la sortance (BAS) n'est pas égale à la sortance (HAUT), comme c'est parfois le cas, on choisit la plus petite des deux valeurs.

EXEMPLE 8-6

Reportez-vous à la fiche technique de l'Appendice II et trouvez le nombre de portes NON-ET 74ALS20 que la sortie d'un autre 74ALS20 peut entraîner.

Solution La fiche technique du 74ALS20 nous fournit les paramètres suivants:

$$I_{OH}(\text{max}) = 0{,}4 \text{ mV} = 400\ \mu\text{A}$$
$$I_{OL}(\text{max}) = 8 \text{ mA}$$
$$I_{IH}(\text{max}) = 20\ \mu\text{A}$$
$$I_{IL}(\text{max}) = 0{,}1 \text{ mA}$$

Analysons en premier le cas du niveau HAUT:

$$\text{Sortance (HAUT)} = \frac{400\ \mu\text{A}}{20\ \mu\text{A}} = 20$$

Maintenant, regardons le cas du niveau BAS:

$$\text{Sortance (BAS)} = \frac{8 \text{ mA}}{0{,}1 \text{ mA}} = 80$$

Pour cet exemple, on doit choisir comme sortance globale 20, puisque c'est la plus petite des deux valeurs. Donc, une sortie d'un 74ALS20 peut piloter jusqu'à 20 entrées de type 74ALS20.

TABLEAU 8-4 FACTEURS DE CHARGE DES SÉRIES TTL

	Charges d'entrée		Sortance	
Séries TTL	**Haut**	**Bas**	**Haut**	**Bas**
74	1 UL	1 UL	10 UL	10 UL
74H	1,25 UL	1,25 UL	12,5 UL	12,5 UL
74L	0,5 UL	≈0,1 UL	10 UL	2,5 UL
74S	1,25 UL	1,24 UL	25 UL	12,5 UL
74LS	0,5 UL	≈0,25 UL	10 UL	5 UL
74AS	0,5 UL	≈0,3 UL	50 UL	12,5 UL
74ALS	0,5 UL	≈0,06 UL	10 UL	5 UL

$$1 \text{ UL} = \begin{cases} 40\ \mu\text{A (HAUT)} \\ 1{,}6 \text{ mA (BAS)} \end{cases}$$

Charges unitaires Certains fabricants spécifient les courants d'entrée et de sortie de leurs dispositifs au moyen d'une unité appelée Charge unitaire (UL), dont la définition est la suivante:

$$\text{1 charge unitaire (UL)} = \begin{cases} 40\ \mu\text{A dans l'état HAUT} \\ 1{,}6 \text{ mA dans l'état BAS} \end{cases}$$

Voilà comment il faut interpréter cette façon d'exprimer le courant: quand on est dans l'état HAUT, une UL signifie la même chose que 40 μA, quand on est dans l'état BAS, une UL vaut 1,6 mA. Par exemple, si pour un CI on spécifie 10 UL pour les deux états, il faut savoir qu'en langage clair cela veut dire:

$$I_{OH}(\text{max}) = 10 \times 40\ \mu\text{A} = 400\ \mu\text{A}$$
$$I_{OL}(\text{max}) = 10 \times 1{,}6 \text{ mA} = 16 \text{ mA}$$

De la même manière, si on précise qu'un CI a 1 UL pour les deux états, cela se traduit par:

$$I_{OH}(\text{max}) = 1 \times 40\ \mu\text{A} = 40\ \mu\text{A}$$
$$I_{OL}(\text{max}) = 1 \times 1{,}6 \text{ mA} = 1{,}6 \text{ mA}$$

Le tableau 8-4 présente les facteurs d'entrée et de sortie types des diverses séries TTL. Ces valeurs sont indicatives, et il ne faut pas se surprendre si elles varient d'un dispositif à l'autre et d'un fabricant à l'autre. La fiche technique du dispositif doit être consultée pour connaître les courants exacts d'un certain CI.

EXEMPLE 8-7

Déterminez les facteurs de charge d'entrée et de sortie de la puce INVERSEUR 7404 (Appendice II).

Solution On trouvera cette information sous la rubrique « INPUT LOADING/FAN-OUT » de la fiche technique 7404. Maintenant, allez à la colonne « 54/74(UL) » pour trouver à la première ligne, 1.0/1.0. Cela signifie que les conditions d'entrée sont 1 UL pour chaque état. Autrement dit, l'entrée 7404 tire un maximum de 40 μA de la source du signal d'entrée placé à l'état HAUT et fournit un maximum de 1,6 mA à travers une entrée de niveau BAS. Ce sont les courants maximaux du cas pessimiste que l'on retrouve dans des conditions extrêmes. En pratique, on pourra mesurer $I_{IH} = 10\ \mu$A et $I_{IL} = 1{,}1$ mA à l'entrée. La plupart des concepteurs utilisent les valeurs pessimistes fournies par le fabricant dans ses fiches techniques.

La deuxième ligne de la colonne « 54/74(UL) » est 20/10. Ceci signifie que la sortie du 7404 a comme valeur nominale 20 UL dans l'état HAUT et 10 UL dans l'état BAS. En d'autres mots, la sortie du 7404 peut *fournir* jusqu'à

$$20 \times 40\ \mu\text{A} = 800\ \mu\text{A}$$

pour piloter des charges quand elle est au niveau HAUT sans que V_{OH} devienne inférieur à $V_{OH}(\text{min}) = 2{,}4$ V et elle peut *consommer* jusqu'à

$$10 \times 1{,}6 \text{ mA} = 16 \text{ mA}$$

dans l'état BAS sans que V_{OL} devienne supérieur à $V_{OL}(\text{max}) = 0{,}4$ V.

EXEMPLE 8-8

Même exemple mais pour le CI 74LS04.

Solution De nouveau, allez à la rubrique « INPUT LOADING/FAN-OUT » mais cette fois-ci à la colonne 54/74LS. Vous relevez pour l'entrée les valeurs nominales 0.5/0.25. Donc le 74LS04 a un facteur de charge d'entrée de 0,5 UL quand il est à l'état HAUT et de 0,25 UL quand il est à l'état BAS.

Les valeurs nominales pour les sorties sont 10/5,0, ce qui signifie que la sortie du 74LS04 peut piloter 10 UL quand il est dans l'état HAUT et 5 UL quand il est dans l'état BAS.

EXEMPLE 8-9

La sortie d'un INVERSEUR 74S04 fournit le signal d'horloge à un registre parallèle constitué de bascules J-K 74S112. Indiquez le nombre maximal de bascules que ce signal d'horloge peut piloter.

Solution On résout les problèmes de ce genre d'abord en déterminant les capacités en charges unitaires d'une sortie 74S04 puis en établissant les exigences d'entrée en charges unitaires de l'entrée *CLK* du 74S112. Nous savons déjà que la sortance du 74S04, d'après sa fiche technique, est 25 UL (HAUT) et 12,5 UL (BAS). La fiche technique du 74S112 montre que les exigences d'entrée de $\overline{CP}$ sont 2,5 UL dans les deux états. Donc le nombre de charges $\overline{CP}$ 74S112 que la sortie 74S04 peut piloter est:

$$\text{N}^{\text{bre}} \text{ de charges} = \frac{\text{valeur nominale de sortie}}{\text{valeur nominale d'entrée}} = \frac{12{,}5 \text{ UL}}{2{,}5 \text{ UL}} = 5$$

Notez que l'on utilise la valeur nominale de sortie état BAS du 7404 puisque cette valeur est plus restrictive que la valeur nominale de sortie état HAUT.

EXEMPLE 8-10

Même exemple que 8-9 mais pour un 74LS04 et un 74LS112.

Solution Le 74LS04 a comme valeurs nominales de sortie 10 UL (HAUT) et 5 UL (BAS), tandis que les entrées du signal d'horloge du 74LS112 exigent 2 UL (HAUT) et 0,5 UL (BAS). Dans ce cas-ci, il nous faut calculer le nombre de charges 74LS112 que le 74LS04 peut piloter dans chaque état, puis prendre la plus petite de ces deux valeurs.

$$\text{nombre de charges (HAUT)} = \frac{10 \text{ UL}}{2 \text{ UL}} = 5$$

$$\text{nombre de charges (BAS)} = \frac{5 \text{ UL}}{0{,}5 \text{ UL}} = 10$$

Donc un 74LS04 peut piloter **5** entrées $\overline{CP}$ d'un 74LS112.

EXEMPLE 8-11

La sortie d'un certain circuit intégré a comme valeurs nominales $I_{OH}(\text{max}) = 800\ \mu\text{A}$ et $I_{OL}(\text{max}) = 48$ mA. Exprimez la sortance de ce CI en charges unitaires.

Solution

État HAUT: sortance = 800 μA/40 μA = 20 UL
État BAS: sortance = 49 mA/1,6 mA = 30 UL

EXEMPLE 8-12

L'entrée d'horloge d'une certaine bascule a comme valeurs nominales $I_{IL} = 0{,}8$ mA et $I_{IH} = 10\ \mu$A. Exprimez son exigence d'entrée en charges unitaires.

Solution

État HAUT: exigence d'entrée = 10 μA/40 μA = 0,25 UL
État BAS: exigence d'entrée = 0,8 mA/1,6 mA = 0,5 UL

Ces exemples devraient vous convaincre que l'expression en fonction de UL est simplement une façon commode de spécifier les courant nominaux d'entrée et de sortie d'un CI. Certains fabricants préfèrent donner pour ces courants les valeurs réelles plutôt qu'un équivalent en UL, d'autres ont recours aux charges unitaires; enfin, certains expriment leurs courants dans les deux systèmes. Comme vous êtes appelé à travailler avec des CI provenant de plusieurs fabricants, vous devrez convertir les UL en ampères.

QUESTIONS DE RÉVISION

1. Quels facteurs déterminent la valeur nominale I_{α}(max) d'un dispositif?
2. Une certaine sortie de TTL possède pour ses deux états une sortance de 5 UL. Quelle intensité de courant peut-elle fournir à des charges quand elle est dans l'état HAUT?
3. Que peut-il arriver à une sortie TTL raccordée à plus de charges unitaires que sa valeur nominale spécifiée?
4. Combien d'entrées $\overline{CP}$ 74S112 peuvent être pilotées par une sortie 74LS04?

8-6 AUTRES CARACTÉRISTIQUES TTL

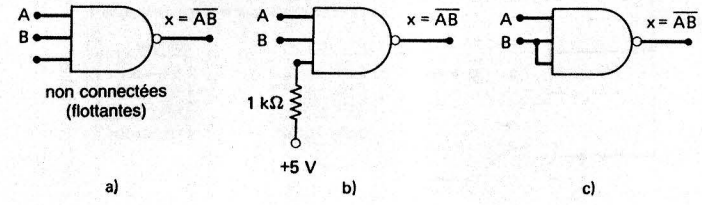
Pour utiliser intelligemment les circuits TTL dans une application numérique, il est nécessaire de connaître d'autres caractéristiques de la logique TTL.

Entrées non connectées (flottantes) Toute entrée d'un circuit TTL non connectée (laissée ouverte) se comporte comme si la valeur qui lui est appliquée était un 1 logique, parce que, dans un cas comme dans l'autre, la jonction émetteur-base ou la diode à l'entrée n'est pas polarisée en direct. Par conséquent, dans *n'importe quel* CI TTL, *toutes* les entrées sont à 1 si elles ne sont pas raccordées à un certain signal logique ou à la masse. Quand une entrée n'est pas connectée, on dit qu'elle est « flottante ».

Entrées inutilisées Souvent, il arrive que toutes les entrées d'un boîtier TTL ne soient pas utilisées dans une certaine application. C'est le cas quand toutes les entrées d'une porte logique ne sont pas nécessaires pour réaliser la fonction logique matérialisée. Par exemple, supposons que nous voulions réaliser la fonction logique $\overline{AB}$ et que pour ce faire on utilise un boîtier contenant une porte NON-ET à trois entrées. Les trois façons possibles de réaliser cette fonction sont montrées à la figure 8-15.

Dans le schéma de la figure 8-15 a), l'entrée inutilisée est flottante et se comporte comme une entrée ayant un 1 logique. La sortie de la porte NON-ET est donc $x = \overline{A \cdot B \cdot 1} = \overline{A \cdot B}$, soit le résultat recherché. Bien que la réponse soit correcte, il est généralement peu souhaitable de laisser une entrée non connectée parce qu'elle devient comme une sorte d'antenne en mesure de capter les signaux parasites rayonnés pouvant entraver le bon fonctionnement de la porte. Il est plus prudent de recourir à la solution de la figure 8-15 b). Dans ce cas-ci, l'entrée inutilisée est raccordée à +5 V à travers une résistance de 1 kΩ, de façon à placer un niveau logique 1 sur cette entrée. La résistance de 1 kΩ a simplement pour rôle de protéger les jonctions émetteur-base des entrées contre les forts courants que peuvent provoquer les surtensions dans la ligne d'alimentation. On peut utiliser la même technique pour les portes ET, puisqu'on ne change pas la sortie si

FIGURE 8-15 Trois façons de prendre en compte des entrées logiques inutilisées.



on place un 1 sur l'entrée inutilisée. Il est possible de raccorder jusqu'à 30 entrées inutilisées à la même résistance de 1 kΩ reliée à V_{CC}.

Une troisième façon de procéder est illustrée à la figure 8-15 c), où l'entrée inutilisée est réunie à une entrée utilisée. C'est une technique satisfaisante en autant que le circuit qui pilote l'entrée *B* ne se retrouve pas ainsi branché à un nombre d'entrées supérieur à sa sortance. C'est une technique qui convient à *n'importe quelle* porte.

Dans le cas des portes OU et NI, il serait inacceptable de laisser les entrées inutilisées déconnectées ou réunies à +5 V, car on se retrouverait avec un niveau de sortie constant (1 pour OU et 0 pour NI) et cela, quels que soient les niveaux des autres entrées. Dans le cas de ces portes, il faut raccorder les entrées inutilisées à la masse (0 V) pour placer sur celle-ci un 0 logique ou les réunir à une entrée utilisée comme à la figure 8-15 c).

Entrées réunies Quand sont réunies deux ou plusieurs entrées de porte TTL afin de constituer une seule entrée, comme à la figure 8-15 c), cette entrée commune a généralement un facteur de charge d'entrée égal à la somme des facteurs de charge individuels. La seule exception concerne les portes NON-ET et ET. Pour ces portes, le facteur de charge d'entrée état BAS sera *le même que celui d'une entrée seule*, quel que soit le nombre d'entrées réunies.

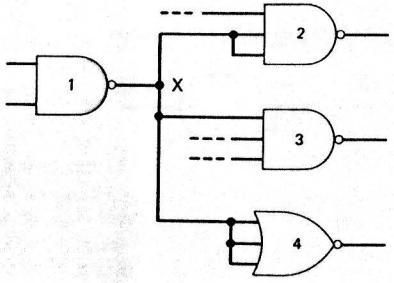
À titre d'exemple, supposons que chacune des trois entrées d'une porte NON-ET, figure 8-15 c), ait comme valeur nominale 1 UL pour chaque état. La borne commune *B* possède donc un facteur de charge d'entrée de 2 UL dans l'état HAUT, mais seulement de 1 UL dans l'état BAS. Le même raisonnement est vrai pour la porte ET. Dans le cas d'une porte OU ou d'une porte NI, l'entrée commune *B* offre un facteur de charge d'entrée de 2 UL dans les deux états.

Pour expliquer cette caractéristique, retournons au schéma de circuit de la porte NON-ET TTL de la figure 8-8 b). Le courant I_{IL} est limité par la résistance R_1. Même si les entrées *A* et *B* sont réunies et mises à la masse, cela ne change pas le courant; il est seulement fractionné entre deux branches parallèles formées par les diodes D_2 et D_3. La situation est toute autre dans le cas des portes OU et NI, parce qu'on ne retrouve pas dans celles-ci des transistors à émetteur multiple, mais plutôt des transistors distincts pour chaque entrée.

EXEMPLE 8-13

Déterminez le nombre de charges unitaires (UL) que la sortie *X* de la figure 8-16 peut piloter. Supposez que chaque entrée de porte a comme valeur nominale 1 UL pour chaque état.

FIGURE 8-16 Exemple 8-16.



facteur de charge sur la sortie de la porte 1

état haut	état bas
2 UL – porte 2	1 UL – porte 2
1 UL – porte 3	1 UL – porte 3
3 UL – porte 4	3 UL – porte 4
6 UL – total	5 UL – total

Solution Le facteur de charge, soit le nombre de portes que l'on peut raccorder à *X*, diffère selon que sa sortie est 0 ou 1, comme on peut le voir sur la figure. Remarquez que les entrées réunies de la porte NON-ET 2 ne comptent que pour 1 UL dans l'état BAS. Les entrées de la porte NI sont considérées comme des charges distinctes dans un état comme dans l'autre.

Mise à 0 des entrées TTL Parfois, certaines situations exigent qu'une entrée TTL soit gardée normalement au niveau BAS puis amenée au niveau HAUT par le déclenchement d'un interrupteur mécanique. Une telle situation est illustrée à la figure 8-17 dans le cas de l'entrée d'un monostable. Ce monostable est déclenché par le front montant qui apparaît lorsque l'interrupteur est fermé momentanément. La résistance *R* sert à garder l'entrée *T* au niveau BAS quand l'interrupteur est ouvert. Il faut prendre bien soin de choisir la valeur de *R* de façon à ce que la différence de potentiel qui apparaît entre ses bornes, quand elle est parcourue par le courant I_{IL} qui quitte l'entrée du MONOSTABLE vers la masse, ne dépasse pas V_{IL}(max). Donc la valeur la plus élevée que peut avoir *R* est:

$$I_{IL} \times R_{max} = V_{IL}(\text{max})$$

$$R_{max} = \frac{V_{IL}(\text{max})}{I_{IL}} \qquad (8\text{-}3)$$

R doit donc être inférieur à cette valeur si on veut être certain que l'entrée du MONOSTABLE est bel et bien au niveau BAS quand l'interrupteur est ouvert. La valeur minimale de *R* est calculée d'après le courant débité par l'alimentation de 5 V quand l'interrupteur est fermé. En pratique, le courant extrait doit être minimisé en s'assurant que *R* est légèrement inférieur à R_{max}.

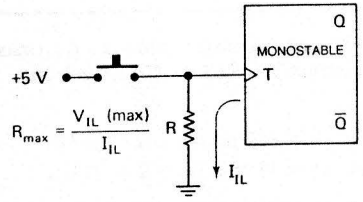


FIGURE 8-17

EXEMPLE 8-14

Trouvez une valeur acceptable pour *R* si le monostable est un CI TTL standard dont le facteur de charge nominale d'entrée est 1 UL.

Solution Puisqu'on a 1 UL, la valeur maximale de I_{IL} est 1,6 mA. C'est cette valeur qu'on utilise pour calculer R_{max}. Dans le cas d'un TTL standard, V_{IL}(max) = 0,8 V. Nous avons donc

$$R_{max} = \frac{0{,}8\ \text{V}}{1{,}6\ \text{mA}} = 500\ \Omega$$

Une valeur acceptable pourrait donc être $R = 470\ \Omega$, une valeur de résistance normalisée.

EXEMPLE 8-15

Même question que l'exemple 8-14, mais pour le cas où le monostable fait partie de la série LS-TTL dont le facteur de charge d'entrée nominale est 0,25 UL.

Solution Avec 0,25 UL, $I_{IL}(\text{max}) = 0{,}4$ mA. Pour un circuit LS-TTL, $V_{IL}(\text{max}) = 0{,}8$ V. Donc

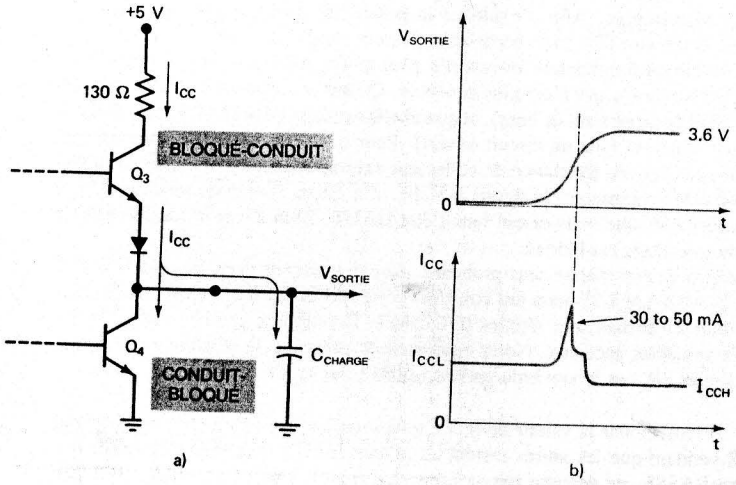
$$R_{\text{max}} = \frac{0{,}8 \text{ V}}{0{,}4 \text{ ma}} = 2 \text{ k}\Omega$$

Une bonne valeur pour la résistance pourrait être 1,8 kΩ, une valeur normalisée.

Courants transitoires À cause de leur sortie en totem, les circuits logiques TTL sont affectés par des courants transitoires générés dans les circuits. Quand la sortie commute de l'état BAS à l'état HAUT (voir figure 8-18), les deux transistors de sortie changent d'état; Q_3 passe de l'état BLOQUÉ à l'état CONDUCTEUR et Q_4 passe de l'état CONDUCTEUR à l'état BLOQUÉ. Comme l'état de départ de Q_4 est la saturation, il lui faut plus de temps pour commuter que Q_3. Il y a donc un court délai (environ 2 ns) pendant lequel les deux transistors sont conducteurs et pendant lequel l'alimentation de +5 V délivre un sursaut de courant relativement intense (30-50 mA). La durée de ce courant transitoire est prolongée par toute capacité de charge sur la sortie du circuit. Cette capacité est la capacité parasite du câblage et la capacité d'entrée de tout circuit de charge. Ces condensateurs parasites doivent être chargés jusqu'à la tension de sortie état HAUT. On peut résumer ainsi cette caractéristique: *quand une sortie TTL en totem passe de BAS à HAUT, une pointe de courant de grande intensité est prélevée de l'alimentation* V_{CC}.

Dans un circuit ou un système numérique complexe, il peut y avoir en même temps plusieurs sorties TTL qui permutent, chaque permutation de l'alimentation drainant une pointe étroite de courant. L'effet cumulatif de toutes ces pointes de courant peut produire

FIGURE 8-18 Une pointe de courant intense est extraite de l'alimentation V_{cc} quand une sortie en totem commute de BAS à HAUT.



une pointe de tension sur la ligne commune V_{CC}, en raison, principalement, de l'inductance répartie de la ligne d'alimentation [se souvenir que $V = L(di/dt)$ pour une inductance et que di/dt est très grand dans le cas d'une pointe de courant de 2 ns]. Une pointe de tension de ce genre peut être la cause de sérieux problèmes de fonctionnement durant les commutations à moins d'utiliser une sorte de filtre. La technique la plus courante est d'utiliser des condensateurs RF raccordés entre V_{CC} et la MASSE pour, à toutes fins pratiques, « court-circuiter » ces pointes de haute fréquence. On parle de la technique du **découplage des alimentations**.

En pratique, il est courant de raccorder un condensateur céramique plat de faible inductance et de capacité 0,01 μF ou 0,1 μF, entre V_{CC} et la masse pour chaque CI TTL enfiché dans une carte de circuit. Les fils de ce condensateur sont les plus petits possibles afin de minimiser l'inductance.

En outre, il est aussi courant en pratique de connecter un seul gros condensateur (2 à 20 μF) entre V_{CC} et la masse sur chaque carte afin de filtrer les variations éventuelles de V_{CC} imputables aux grands changements des niveaux I_{CC}, quand les sorties commutent.

QUESTIONS DE RÉVISION

1. Dites quelle est la sortie d'une porte NON-ET TTL quand toutes ses entrées ne sont pas connectées.
2. Dites quelles sont les deux façons acceptables de traiter les entrées inutilisées d'une porte ET.
3. Même question mais dans le cas de la porte NI.
4. *Vrai ou faux:* Quand des entrées d'une porte NON-ET sont réunies, elles sont toujours considérées comme une seule charge du point de vue de la source du signal.
5. Définissez découplage des alimentations. À quoi sert-il?

8-7 SORTIES TTL À COLLECTEUR OUVERT

Examinons le circuit logique de la figure 8-19 a). La combinaison des portes NON-ET 4 et 5 réalise la fonction ET qui multiplie logiquement les sorties des portes NON-ET 1, 2 et 3; l'expression de la sortie finale X est alors:

$$X = \overline{AB} \cdot \overline{CD} \cdot \overline{EF}$$

Le circuit de la figure 8-19 b) montre que la même opération logique est obtenue simplement en raccordant les sorties des portes NON-ET 1, 2 et 3. Autrement dit, le fait de raccorder ensemble les sorties réalise l'opération ET. Voici comment on explique cela: quand les sorties sont réunies et que l'une des sorties passe au niveau BAS, le point commun doit passer au niveau BAS puisque le transistor Q_4 de la sortie mise à zéro « court-circuite à la masse » ce point. La sortie commune est donc au niveau HAUT seulement quand toutes les sorties sont au niveau HAUT, ce qui, comme vous le savez déjà, est la définition de l'opération ET.

Le montage de la figure 8-19 b) présente un avantage par rapport au montage courant de la figure 8-19 a): il lui faut moins de portes pour produire la même expression de sortie. On donne le nom de **ET câblé** à un montage de cette nature, parce que la réunion des sorties réalise l'opération ET. Parfois, vous verrez ce montage appelé à tort *OU câblé*.

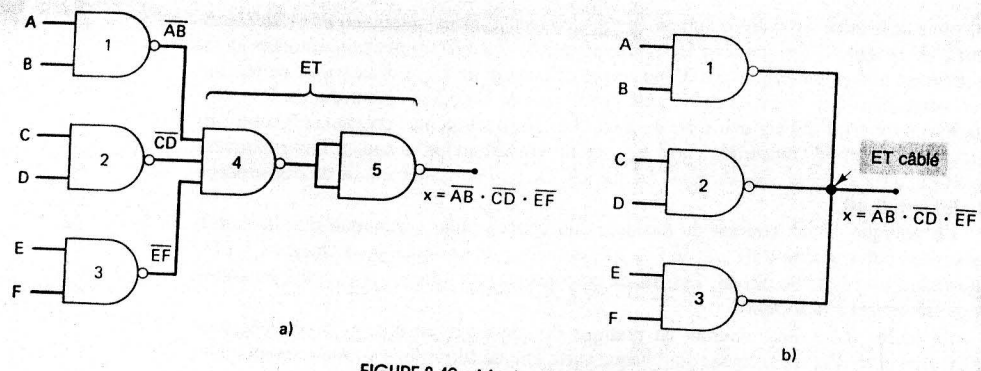
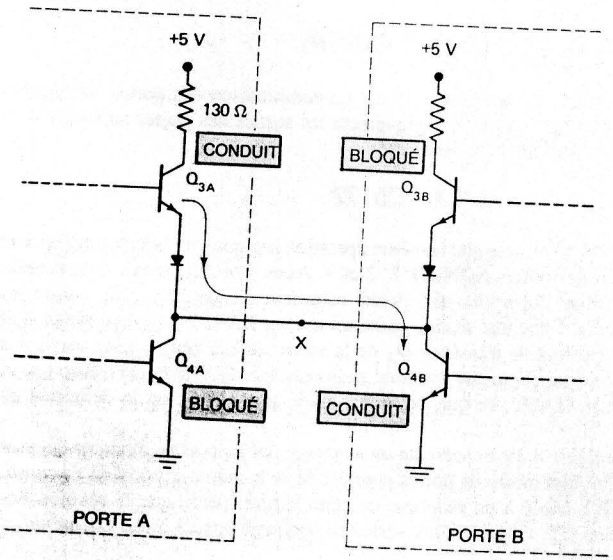


FIGURE 8-19 Mode d'action du ET câblé.

ET câblé irréalisable avec une sortie en totem Pour pouvoir tirer avantage du montage ET câblé, il faut pouvoir réunir deux ou plusieurs portes sans conséquences néfastes. Malheureusement, les circuits TTL classiques, dont la sortie est en totem, ne peuvent pas être réunis de cette façon. C'est ce qu'illustre la figure 8-20 où les sorties en totem de 2 portes distinctes sont réunies au point X. Supposez que la sortie de la porte A soit dans l'état HAUT (Q_{3A} conduit, Q_{4A} bloqué) et que la sortie de la porte B soit dans l'état BAS (Q_{3B} bloqué, Q_{4B} conduit). Dans cette situation, Q_{4B} est une charge de très faible résistance pour Q_{3A}, ce qui draine un courant pouvant atteindre 55 mA. Un tel courant peut facilement endommager Q_{3A} ou Q_{4B}. La situation est encore pire quand on réunit plus de deux sorties TTL.

FIGURE 8-20 La réunion de sorties en totem peut donner lieu à un courant trop intense à travers Q_4.



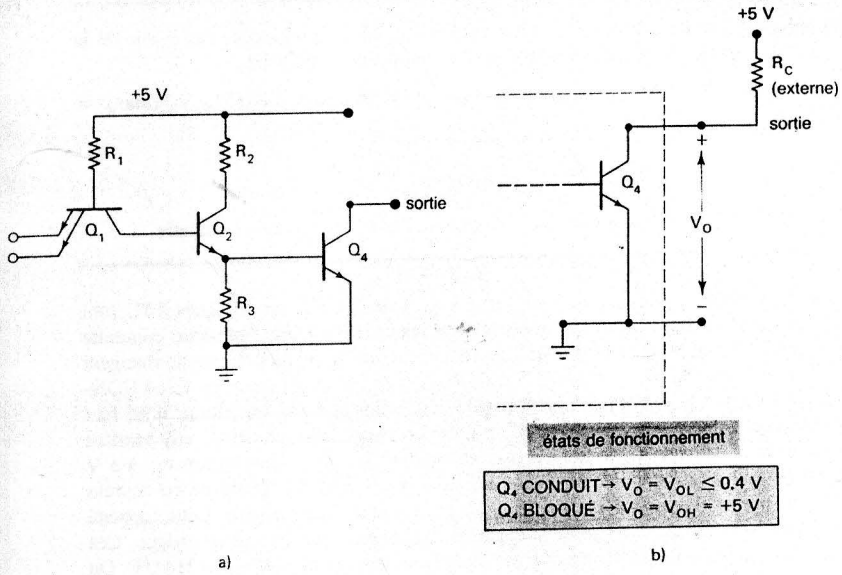


FIGURE 8-21 a) Circuit TTL à collecteur ouvert; b) résistance externe de rappel à la source.

Des sorties en totem sont parfois réunies par inadvertance à cause d'erreurs de câblage ou de court-circuitages accidentels sur une carte de circuit imprimé. Quand cela se produit, le signal qui paraît au point commun est généralement la multiplication logique des deux signaux de sortie ainsi réunis. Toutefois, cela n'est vrai que pour les circuits TTL; les circuits MOS et CMOS réagissent différemment quand leurs sorties sont réunies.

Sorties à collecteur ouvert Afin d'exploiter la possibilité du ET câblé, les fabricants ont conçu quelques circuits TTL avec sorties à *collecteur ouvert*. Comme le schématise la figure 8-21 a), le circuit à collecteur ouvert n'a plus ni Q_3, ni D_1, ni R_4. La sortie est prélevée au collecteur de Q_4 qui n'est plus connecté. Quand la sortie est à l'état BAS, Q_4 est CONDUCTEUR (a un courant de base), et quand elle est dans l'état HAUT, ce transistor est BLOQUÉ (est dans les faits un circuit ouvert). Pour qu'un tel circuit fonctionne correctement, il faut ajouter une résistance de collecteur externe R_c, dite résistance de rappel à la source, connectée comme sur la figure 8-21 b), afin qu'un niveau de tension HAUT apparaisse à la sortie lorsque celle-ci est dans l'état HAUT. Sans R_c, il n'y aurait pas de tension de sortie quand Q_4 est bloqué.

Le ET câblé peut être réalisé sans problème avec des circuits dont les sorties sont à collecteur ouvert. La figure 8-22 nous fait voir trois portes NON-ET à 2 entrées à *collecteur ouvert* (7401) dont les sorties sont réunies (ET câblé). Remarquez que ces portes NON-ET n'ont pas de symboles spéciaux. Notez également la présence de la résistance externe de rappel à la source (R_c) et le symbole parfois utilisé pour le ET câblé.

Valeur de R_c Il faut choisir la valeur de R_c de telle manière que lorsqu'une sortie passe au niveau BAS pendant que les autres restent au niveau HAUT, le courant absorbé, qui arrive à la sortie BASSE, ne dépasse pas sa limite I_{OL}(max). Une valeur de $R_c = 1$ kΩ donne un courant absorbé de 5 mA à travers le transistor de sortie de la porte au niveau

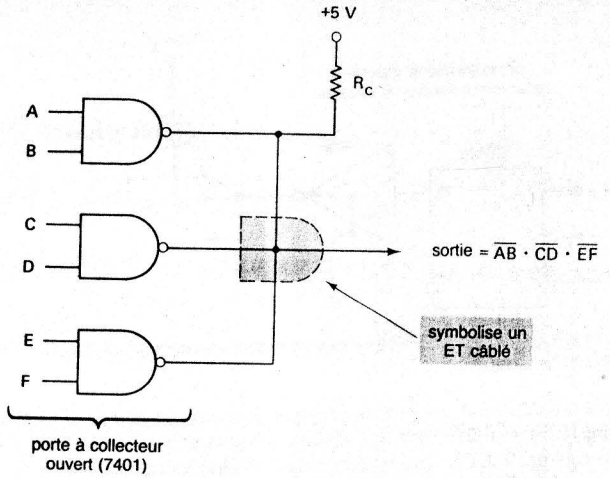


FIGURE 8-22 ET câblé réalisé avec des portes à collecteur ouvert.

BAS. Comme cette sortie pilote généralement d'autres charges TTL dont les courants absorbés s'additionnent, il ne faut pas que cette somme dépasse $I_{OL}(\text{max})$. Il peut donc sembler qu'une valeur élevée de R_c serait souhaitable. Toutefois, il faut être bien conscient que tout condensateur de charge sera chargé à travers cette résistance R_c; c'est pourquoi il faut aussi qu'elle soit la plus petite possible afin d'améliorer la vitesse de commutation. Le montage à collecteur ouvert, même en utilisant une R_c minimale, est beaucoup plus lent que les sorties TTL en totem, dans lesquelles Q_3 fonctionne comme un émetteur-suiveur de basse impédance pour charger les condensateurs de charge. C'est la raison pour laquelle il ne faut pas utiliser de circuits à collecteur ouvert dans les applications où la rapidité de commutation est un élément essentiel.

EXEMPLE 8-16

Le CI 7405 (voir Appendice II) renferme six inverseurs dont les sorties sont à collecteur ouvert. Ces six inverseurs sont montés de façon à réaliser un ET câblé, voir figure 8-23 a).

a) Déterminez l'expression logique de la sortie X.

b) Calculez R_c en supposant que la sortie X doit piloter d'autres circuits dont le facteur de charge totale est 4 UL.

Solution

a) La sortie de chaque inverseur est l'inverse de son entrée. Le ET câblé a pour effet simplement de mettre en relation ET chacune des sorties des inverseurs. Donc

$$X = \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} \cdot \overline{E} \cdot \overline{F}$$

En vertu du théorème de DeMorgan, on sait que cette dernière expression est équivalente à

$$X = \overline{A + B + C + D + E + F}$$

Ce qui correspond à l'opération NI.

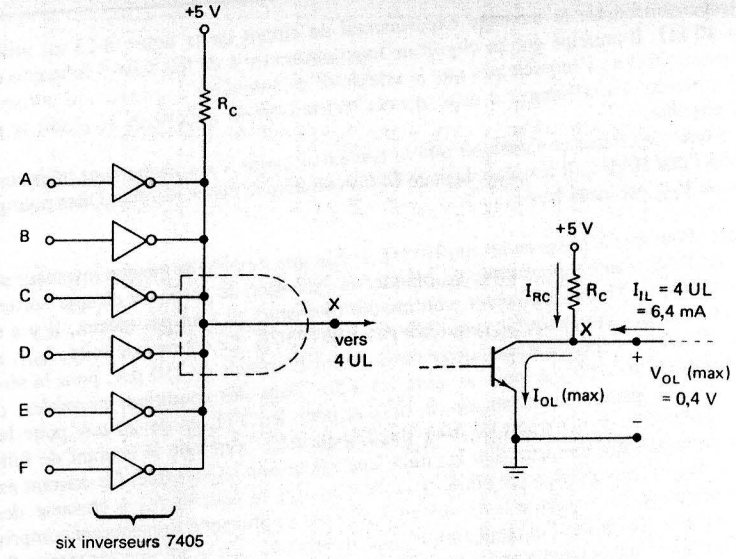


FIGURE 8-23 Exemple 8-11.

b) Si on suppose qu'un seul des INVERSEURS a une entrée au niveau HAUT, c'est le transistor de sortie de cet INVERSEUR qui écoulera les courants I_{RC} et I_{IL}, comme on peut le voir à la figure 8-23 b). Si on se reporte à la fiche technique du 7405, on s'aperçoit qu'il a une sortance de 10 lorsqu'il est au niveau BAS. Donc il peut consommer un courant total de $I_{OL}(\text{max}) = 16$ mA. En additionnant tous les courants qui arrivent à X, on trouve

$$I_{OL}(\text{max}) = I_{RC} + I_{IL}$$

$$16 \text{ mA} = I_{RC} + 6{,}4 \text{ mA}$$

$$\therefore I_{RC} = 9{,}6 \text{ mA}$$

La tension aux bornes de R_c sera $V_{CC} - V_{OL}(\text{max})$; nous utilisons $V_{OL}(\text{max})$ étant donné que ce transistor absorbe son courant maximal. Dans le cas des TTL standard, nous savons que $V_{OL}(\text{max}) = 0{,}4$ V. Donc, nous obtenons

$$R_c(\text{min}) = \frac{V_{CC} - V_{OL}(\text{max})}{I_{RC}} = \frac{5 \text{ V} - 0{,}4 \text{ V}}{9{,}6 \text{ mA}} = 480\ \Omega$$

En pratique, la valeur utilisée est légèrement supérieure à celle-ci afin d'avoir une certaine marge de sécurité. Souvenez-vous que nous voulons que R_c soit le plus petit possible afin d'éviter d'accroître le temps de commutation. Un bon choix pourrait être la valeur normalisée de 560 Ω.

EXEMPLE 8-17

Un technicien étudie le montage expérimental du circuit de la figure 8-23 en utilisant $R_c = 10\ k\Omega$. Il anticipe que ce circuit ne fonctionnera qu'à de très basses fréquences, et c'est pourquoi il ne s'inquiète pas que la vitesse de commutation augmente s'il utilise une R_c plus élevée. Parallèlement à cela, il veut que le courant prélevé de V_{CC} soit le plus faible possible.

Il teste ce circuit et s'aperçoit que la tension au point X est légèrement inférieure à 2 V dans l'état HAUT. Il remplace la puce 7405 et observe le même résultat. Dites pourquoi la tension V_{CC} est aussi basse.

Solution Pour que X passe au niveau HAUT, il faut que l'entrée de chaque inverseur soit au niveau BAS. Ceci a pour effet de bloquer le transistor de sortie de chaque sortie à collecteur ouvert. Malheureusement, comme ces transistors ne sont pas idéaux, il y a un certain courant de fuite lorsqu'ils sont dans l'état BLOQUÉ. Ce courant de fuite, I_{OH}, ne doit pas, d'après les spécifications, dépasser la *valeur maximale* de 250 μA, pour la série TTL standard et de 100 μA pour la série LS-TTL. Dans des conditions *normales*, ce courant se situe généralement autour de 50 μA pour les TTL et de 20 μA pour les LS-TTL. Toutefois, ces courants de fuite s'additionnent de sorte que le courant de fuite total des six transistors de sortie peut atteindre une valeur non négligeable. Ce courant est fourni par V_{CC} à travers R_c. En outre, V_{CC} doit fournir le courant I_{IH} à chacune des 4 charges unitaires. Par conséquent, il n'est peut-être pas surprenant qu'un courant compris entre 300 et 400 μA quitte l'alimentation V_{CC} et traverse R_c. En ayant une résistance $R_c = 10\ k\Omega$, il n'est plus étonnant qu'il y ait une chute de tension d'environ 3 V dans cette résistance, ce qui ne laisse que 2 V au point X.

Parce que les courants de fuite des sorties à collecteur ouvert et le courant I_{IH} des charges traversent R_c, il est préférable de prendre comme valeur pour R_c la valeur minimale calculée en se fondant sur le courant d'absorption maximal de la sortie à collecteur ouvert, soit I_{OL}(max).

Tampons/pilotes à collecteur ouvert Les circuits logiques appelés *tampons, pilotes ou tampons/pilotes* sont conçus pour avoir un courant de sortie plus intense ou la capacité de fournir une tension supérieure à celle d'un circuit logique ordinaire. Les circuits intégrés tampons/pilotes que l'on trouve dans le commerce viennent soit avec des sorties en totem soit avec des sorties à collecteur ouvert.

Le boîtier 7406 est un CI tampon/pilote à collecteur ouvert très apprécié qui renferme six INVERSEURS (sorties à collecteur ouvert) absorbant jusqu'à 40 mA dans l'état BAS. En outre, ce boîtier peut présenter sur ses sorties des tensions allant jusqu'à 30 V. Cela veut donc dire que le transistor de sortie peut être raccordé à une tension supérieure à 5 V.

Un circuit dans lequel un boîtier 7406 sert de tampon entre une bascule 74LS112 et une lampe incandescente ayant comme valeurs nominales 24 V, 25 mA est illustré à la figure 8-24. Le 7406 allume ou éteint la lampe conformément à l'état de la sortie Q de la bascule. Notez que cette lampe est alimentée par une tension de +24 V et qu'elle joue le rôle de la résistance de rappel à la source pour la sortie à collecteur ouvert.

Quand $Q = 1$, la sortie du 7406 passe au niveau BAS et son transistor de sortie absorbe le courant de la lampe de 25 mA fourni par la source, ce qui fait BRILLER cette lampe. Quand $Q = 0$, le transistor de sortie du 7406 est BLOQUÉ; il n'y a donc pas de trajet pour le courant et la lampe est ÉTEINTE. Dans cet état, la tension de 24 V se retrouve aux bornes du transistor de sortie BLOQUÉ de sorte que $V_{OH} = 24$ V, une valeur inférieure à la valeur nominale maximale V_{OH} du 7406.

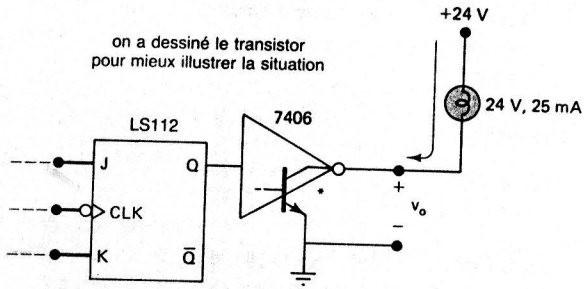


FIGURE 8-24 Un tampon/pilote à collecteur ouvert excite une charge de haute intensité et de haute tension.

Les sorties à collecteur ouvert servent fréquemment comme pilotes de voyant DEL, ainsi que l'illustre la figure 8-25. Dans ce schéma, on peut utiliser un 7405 ou un 7406, selon le courant exigé par la DEL. La résistance sert à limiter le courant à une valeur sûre. Quand la sortie de l'INVERSEUR est BASSE, son transistor de sortie offre un chemin de faible résistance vers la masse au courant de la DEL, de sorte que celle-ci brille. Quand la sortie de l'INVERSEUR est HAUTE, son transistor de sortie ne conduit pas, et il n'y a pas de courant dans la DEL; celle-ci est éteinte.

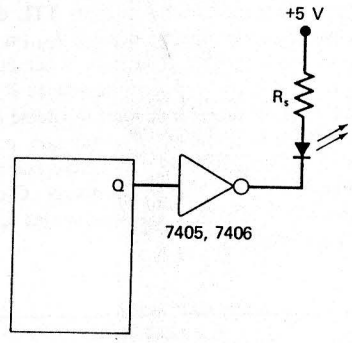
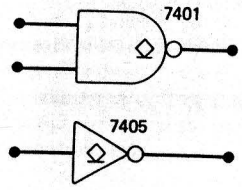


FIGURE 8-25 Une sortie à collecteur ouvert peut servir à piloter un voyant DEL.

Symbole IEEE/ANSI pour les sorties à collecteur ouvert Dans les symboles traditionnels des circuits logiques, rien ne distingue une sortie à collecteur ouvert d'une sortie en totem. Par contre, la nouvelle symbologie IEEE/ANSI fait la distinction et utilise une notation spéciale pour indiquer une sortie à collecteur ouvert. La figure 8-26 illustre le signe standard

FIGURE 8-26 Notation IEEE/ANSI pour des sorties à collecteur ouvert.



IEEE/ANSI employé pour signaler une sortie à collecteur ouvert. Il s'agit d'un losange souligné. Bien que nous ne nous conformions pas à la symbologie IEEE/ANSI dans ce livre, nous choisissons de conserver le losange souligné pour désigner dans nos symboles traditionnels les sorties à collecteur ouvert.

QUESTIONS DE RÉVISION

1. Qu'est-ce qu'un ET câblé?
2. Pourquoi les éléments logiques à sortie en totem ne peuvent pas servir à former un ET câblé?
3. En quoi les sorties à collecteur ouvert diffèrent-elles des sorties en totem?
4. Pourquoi une sortie à collecteur ouvert a-t-elle besoin d'une résistance de rappel à la source?
5. Quels facteurs interviennent dans la détermination de la valeur d'une résistance de rappel à la source?
6. Pourquoi les sorties à collecteur ouvert sont-elles généralement plus lentes que les sorties en totem?
7. Quel est le symbole IEEE/ANSI d'une sortie à collecteur ouvert?

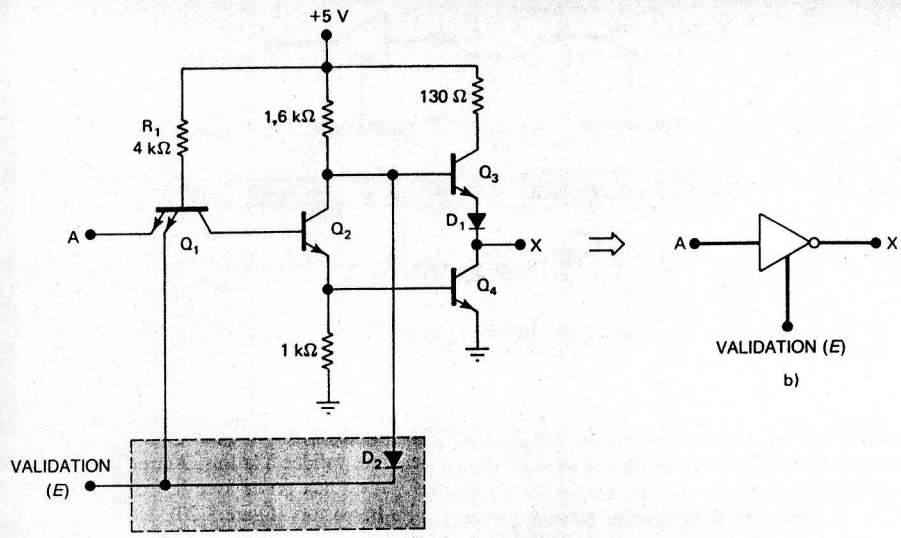
8-8 TTL TROIS ÉTATS

Il s'agit d'un troisième type de configuration de la sortie pour la logique transistor-transistor. Celui-ci est aussi rapide que le montage en totem mais peut avoir, en plus, ses sorties branchées en parallèle. Ce type de montage s'appelle TTL trois états parce qu'il possède *trois* états de sortie possibles: HAUT, BAS et HAUTE IMPÉDANCE. L'état HAUTE IMPÉDANCE est la condition dans laquelle se trouve le circuit quand les deux transistors du montage en totem sont BLOQUÉS, de sorte que la borne de sortie présente une haute impédance par rapport à la masse et à V_{CC}. Autrement dit, la sortie est une borne ouverte ou flottante qui n'est ni au niveau BAS ni au niveau HAUT. En pratique, cette borne de sortie n'est pas exactement un circuit ouvert, mais sa résistance par rapport à la masse et à V_{CC} est de quelques mégohms.

Le circuit qui manifeste ce fonctionnement à trois états est une version modifiée du circuit en totem de base. La figure 8-27 a) nous fait voir le circuit d'un INVERSEUR trois états; on peut voir sur ce circuit une nouvelle section par rapport au circuit de base, celle entourée de lignes tiretées. Ce circuit a deux entrées: A, l'entrée logique normale et E, l'entrée VALIDATION qui produit l'état HAUTE IMPÉDANCE. Examinons le fonctionnement pour les deux états de E.

L'état validé Quand $E = 1$, le circuit fonctionne comme un inverseur normal puisqu'une tension au niveau HAUT appliquée sur E n'a aucun effet sur Q_1 ou D_2. Dans la condition validée, la sortie est simplement l'inverse de l'entrée logique A.

L'état invalidé (haute impédance) Quand $E = 0$, le circuit se place dans son état HAUTE IMPÉDANCE et y reste quel que soit le niveau logique sur l'entrée A. Le signal BAS sur E polarise en direct la jonction émetteur-base de Q_1 et dérive le courant de R_1 qui allait sur la base de Q_2 de sorte que Q_2 se retrouve BLOQUÉ, ce qui en retour bloque Q_4. Le signal BAS sur E polarise également en direct la diode D_2 pour dériver vers la masse le courant qui allait sur la base de Q_3, de sorte que ce dernier transistor est aussi BLOQUÉ.



conditions de fonctionnement

VALIDATION (*E*)	SORTIE
HAUT	Validé: fonctionne comme un INVERSEUR
BAS	Invalidé: la sortie est dans un état HAUTE IMPÉDANCE. L'entrée A n'a aucun effet.

c)

FIGURE 8-27 INVERSEUR TTL trois états.

Quand les deux transistors du totem sont dans leur état non conducteur, la borne de sortie est essentiellement un circuit ouvert. C'est ce qu'on illustre symboliquement dans le tableau de la figure 8-27 c).

Le symbole logique d'un INVERSEUR trois états est illustré à la figure 8-27 b). Remarquez que l'entrée VALIDATION est placée dans le symbole de l'INVERSEUR. Notez également que E est vrai au niveau HAUT, c'est-à-dire que l'INVERSEUR est validé quand $E = 1$.

Avantage du circuit trois états Les sorties des CI trois états peuvent être réunies (montées en parallèle) sans que ne diminue la vitesse de commutation. Cela est possible parce que lorsqu'une sortie trois états est validée, elle fonctionne comme une sortie en totem, et l'on sait qu'une sortie de ce genre a une basse impédance et une grande vitesse. Il y a un fait toutefois qu'il est important de remarquer: quand des sorties trois états sont montées en

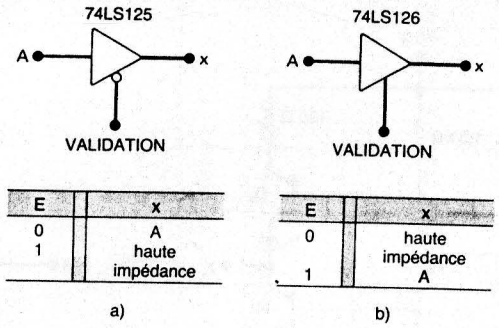


FIGURE 8-28 Tampons trois états.

parallèle, il n'y a qu'une de ces sorties qui peut être validée à la fois. Autrement, si deux sorties en totem actives se retrouvent réunies, on est aux prises avec le même problème de courants trop intenses qui peuvent griller les transistors (figure 8-20). Nous en dirons plus sur ce sujet dans le paragraphe suivant qui traite des tampons trois états.

Tampons trois états Un *tampon trois états* est un circuit dont on se sert pour contrôler le passage d'un signal logique entre une entrée et une sortie. Certains tampons trois états occasionnent l'inversion du signal qui les traverse. Le circuit de la figure 8-27 est pour cette raison appelé *un tampon trois états inverseur*.

Deux des tampons trois états en version CI les plus courants sont les boîtiers 74LS125 et 74LS126. On retrouve dans ces deux boîtiers 4 tampons trois états non inverseurs comme celui montré à la figure 8-28. La seule différence entre le 74LS125 et le 74LS126 est le niveau vrai de l'entrée de VALIDATION. Le boîtier 74LS125 autorise le passage du signal d'entrée A quand $E = 0$, alors que le boîtier 74LS126 autorise ce passage quand $E = 1$.

Les tampons trois états sont très utilisés dans les circuits où plusieurs signaux sont raccordés à des faisceaux de conducteurs communs (bus). Nous étudierons certaines de ces utilisations au chapitre 9. Pour l'instant, la figure 8-29 a) nous donne une première idée de leur utilisation. Dans le circuit schématisé, il y a trois signaux logiques A, B et C raccordés à un bus commun par l'intermédiaire de tampons trois états 74LS126. Ce montage permet de transmettre à d'autres circuits n'importe lequel des signaux en le plaçant sur le bus grâce à la validation du tampon approprié.

Par exemple, considérons la situation de la figure 8-29 b) où $E_B = 1$ et $E_A = E_C = 0$. Cette condition désactive les tampons inférieur et supérieur de sorte que leur sortie est dans l'état haute impédance, ce qui signifie plus simplement qu'ils ne sont pas connectés au bus. C'est ce que veut dire le X sur le schéma. Le tampon du centre est activé, ce qui fait en sorte que sa sortie B franchit la porte et se retrouve sur le bus, qui l'achemine à d'autres circuits également connectés à ce bus. Quand des sorties trois états sont reliés ensemble, comme sur la figure 8-29, il est important de ne pas oublier qu'il ne doit y avoir qu'une sortie de validée à la fois. Dans le cas contraire, on se retrouve avec au moins deux sorties en totem raccordées ensemble, ce qui engendre des courants pouvant être destructeurs. Et même si aucune détérioration ne survient, le signal sur le bus n'a plus de sens puisqu'il est une combinaison de plusieurs signaux. On appelle généralement une situation de ce genre **un conflit de bus**. Dans les bus à trois états, le concepteur doit s'assurer que les signaux de validation ne provoquent pas un conflit de bus.

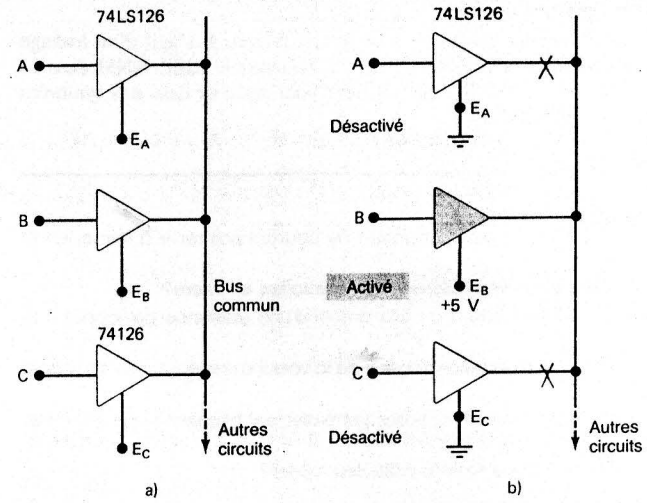
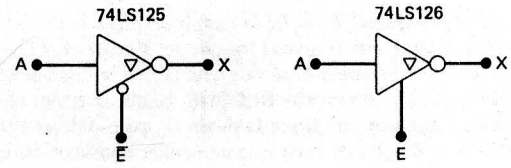


FIGURE 8-29 a) Tampons trois états utilisés pour relier plusieurs signaux à un bus commun; b) conditions assurant la transmission du signal B sur le bus.

CI trois états En plus des tampons trois états, il existe une bonne gamme de circuits intégrés dont les sorties sont à trois états. C'est le cas, par exemple, du 74LS374, un registre à huit bascules D. Plus précisément, il s'agit d'un registre à 8 bits formé de bascules D dont les sorties sont raccordées à des tampons trois états. Un tel registre peut être relié à un faisceau de lignes communes, auquel sont déjà raccordés d'autres dispositifs semblables, afin de permettre l'échange de données entre eux par l'intermédiaire du faisceau (bus). Nous examinerons plus complètement le *bus de données à trois états* au chapitre 9. Parmi les autres dispositifs offerts en version sorties à trois états, mentionnons des décodeurs, des multiplexeurs, des convertisseurs analogiques-numériques, des mémoires et des microprocesseurs.

Symbole IEEE/ANSI des sorties trois états Dans les symboles logiques traditionnels, il n'y a pas de signe distinctif pour marquer une sortie trois états. Par contre, dans la symbologie IEEE/ANSI, il y en a un illustré à la figure 8-30. Il s'agit d'un triangle inversé. Bien que ce signe ne fasse pas partie de la symbologie traditionnelle, nous choisissons de le conserver dans le reste de ce livre.

FIGURE 8-30 Notation IEEE/ANSI pour les sorties trois états



QUESTIONS DE RÉVISION

1. Quelles sont les trois états de sortie possible d'un CI trois états?
2. Dans quel état se trouve une sortie trois états quand elle est désactivée?
3. Que signifie conflit de bus?
4. Quelles conditions sont nécessaires pour que le signal *C* soit transmis sur le bus de la figure 8-29?
5. Quel signe est utilisé dans la norme IEEE/ANSI pour désigner une sortie trois états?

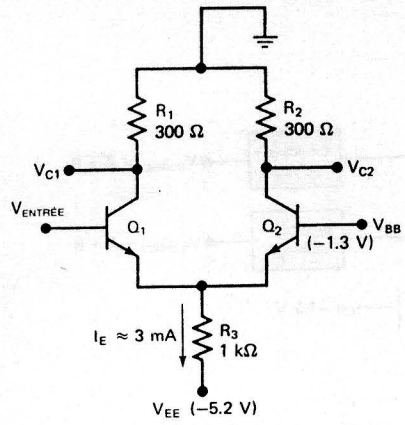
8-9 FAMILLE DE CI NUMÉRIQUES ECL

La famille TTL utilise des transistors fonctionnant en mode saturé. À cause de cela, leur vitesse de commutation est limitée par le retard de stockage associé à un transistor qui est porté à saturation. Une autre famille logique *bipolaire* a été mise au point afin de prévenir la saturation des transistors; ses membres ont donc des vitesses de commutation plus élevées. Cette famille est appelée **logique à couplage par l'émetteur** (ECL), et son principe de fonctionnement est celui de la commutation du courant, qui se produit quand un courant de polarisation fixe inférieur à $I_{c(sat)}$ est transféré du collecteur d'un transistor au collecteur d'un autre transistor. En raison de ce fonctionnement en mode courant, on parle souvent dans le cas de cette famille de **logique en mode courant** (CML).

Circuit ECL de base Le circuit de base de la logique à couplage par l'émetteur est essentiellement le montage de l'amplificateur différentiel de la figure 8-31 a). La tension d'alimentation V_{EE} produit un courant fixe I_E qui demeure autour de 3 mA durant le fonctionnement normal. Ce courant traverse soit Q_1 ou Q_2, selon le niveau de tension sur $V_{ENTRÉE}$. En d'autres mots, ce courant commute entre le collecteur de Q_1 et le collecteur de Q_2 chaque fois que $V_{ENTRÉE}$ passe à son autre niveau logique: $-1{,}7$ V (0 logique pour ECL) et $-0{,}8$ V (1 logique pour ECL). Le tableau de la figure 8-31 a) résume les tensions de sortie pour les deux conditions de $V_{ENTRÉE}$. Il y a deux points importants à remarquer: 1) V_{C1} et V_{C2} sont complémentaires l'un de l'autre, et 2) les niveaux des tensions de sortie ne sont pas les mêmes que les niveaux logiques d'entrée.

La difficulté soulevée par la seconde remarque se résout assez facilement en raccordant V_{C1} et V_{C2} à des émetteurs-suiveurs (Q_3 et Q_4), comme on le montre sur la figure 8-31 b). Ces émetteurs-suiveurs jouent deux rôles: 1) ils retranchent environ 0,8 V de V_{C1} et de V_{C2} pour amener les niveaux de sortie aux niveaux logiques ECL exacts; et 2) ils ont une impédance de sortie très basse (habituellement 7 Ω), ce qui donne une sortance très élevée et une condition qui charge très rapidement les condensateurs parasites. Ce circuit fournit deux sorties complémentaires: $V_{SORTIE\,1}$, qui est égale à $\overline{V_{ENTRÉE}}$ et $V_{SORTIE\,2}$, qui est égale à $V_{ENTRÉE}$.

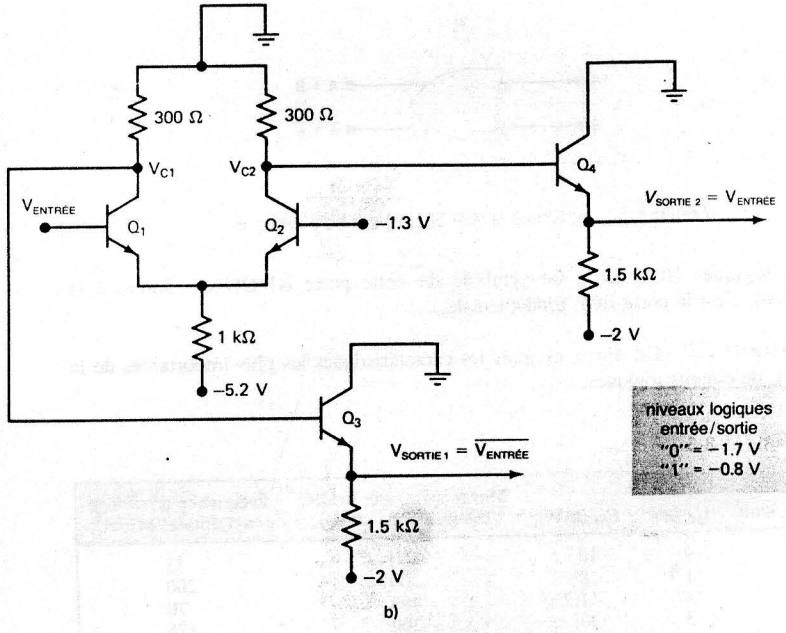
Porte NI/OU ECL Il est possible d'utiliser le circuit ECL de base, illustré à la figure 8-31 b), comme un inverseur si on définit sa sortie comme étant $V_{SORTIE\,1}$. On transforme ce circuit de base pour obtenir plus d'une entrée en montant en parallèle au transistor Q_1 d'autres transistors correspondant aux autres entrées, comme à la figure 8-32 a). Dans ce cas-ci, le courant qui traverse Q_2 peut être dérivé dans Q_1 ou Q_3, et donner lieu à deux sorties $V_{SORTIE\,1}$ et $V_{SORTIE\,2}$ correspondant respectivement aux



a)

États de fonctionnement

$V_{entrée}$	Sorties	
−1,7 V 0 logique	V_{C1} = 0 V V_{C2} = −0,9 V	Q_2 conduit
−0,8 V 1 logique	V_{C1} = −0,9 V V_{C2} = 0 V	Q_1 conduit



b)

FIGURE 8-31 a) Circuit ECL de base; b) addition d'émetteurs-suiveurs.

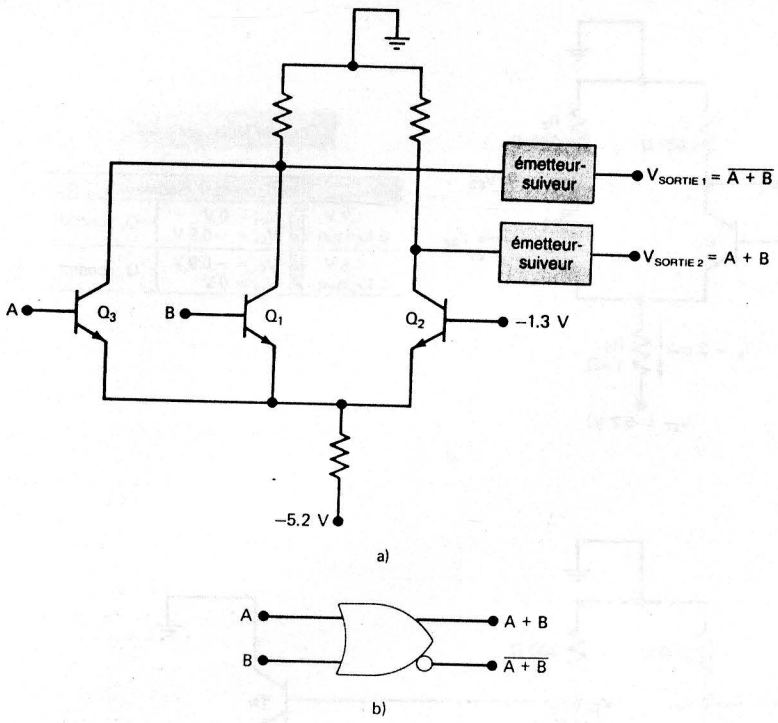


FIGURE 8-32 a) Circuit NI/OU ECL; b) symbole logique.

opérations logiques NI et OU. Le symbole de cette porte NI/OU est donné à la figure 8-32 b); c'est la porte ECL fondamentale.

Caractéristiques ECL On donne ci-après les caractéristiques les plus importantes de la famille ECL de circuits logiques.

TABLEAU 8-5

Famille logique	t_{pd} (ns)	P_D (mW)	Marge sens. aux bruits, cas pessimiste (mV)	Fréquence d'horloge maximale (MHz)
74	9	10	400	35
74AS	1,7	8	300	200
7ALS	4	1,2	400	70
74S	3	20	300	125
74LS	9,5	2	300	45
ECL	1	40	250	600

1. Les transistors ne sont jamais saturés; par conséquent, la vitesse de commutation est très élevée. Le retard de propagation typique est 2 ns, de sorte que les ECL sont un petit peu plus rapides que les TTL Schottky avancés (série 74AS).

2. Les niveaux logiques ont les valeurs nominales −0,8 V et −1,70 V pour les niveaux logiques 1 et 0 respectivement.
3. Les marges de sensibilité aux bruits des ECL pour le cas pessimiste sont environ 250 mV. À cause de ces valeurs, les ECL sont un peu moins fiables lorsqu'ils servent dans des usines de l'industrie lourde.
4. Un bloc logique ECL donne généralement en sortie un état et son complément. Ceci élimine le besoin d'inverseurs.
5. La sortance de ce dispositif est généralement d'environ 25, en raison de la faible impédance des sorties en émetteur-suiveur.
6. La consommation caractéristique d'une porte ECL de base est 40 mW, environ 5 fois la série 74AS.
7. Le courant total qui traverse un circuit ECL est relativement constant quel que soit son état logique. Grâce à cette caractéristique, le courant prélevé de l'alimentation est constant et cela même durant les commutations. C'est donc un circuit qui n'est pas affecté par des pointes parasites comme celles qui se produisent dans les circuits en totem TTL.

Le tableau 8-5 compare les caractéristiques de la famille ECL à celles de la famille TTL.

On ne retrouve pas dans les systèmes les circuits de la famille ECL aussi fréquemment que ceux des familles TTL et MOS, sauf dans les applications à très haute fréquence en raison de leur vitesse plus élevée. Cette famille est désavantagée par rapport aux autres familles logiques à cause de ses marges de sensibilité trop étroites et de sa consommation élevée. Un autre inconvénient qui la touche concerne le recours à des tensions d'alimentation négatives et la valeur de ses niveaux logiques qui ne sont pas compatibles avec ceux des autres familles logiques; c'est une difficulté qui rend presque impossible le mélange dans un même circuit d'éléments ECL, TTL et MOS.

QUESTION DE RÉVISION

1. *Vrai ou faux:*
   a) Les ECL fonctionnent plus rapidement parce que leurs transistors ne se saturent pas.
   b) Les circuits ECL ont deux sorties supplémentaires.
   c) Les marges de sensibilité aux bruits des circuits ECL sont plus larges que les marges de sensibilité des TTL.
   d) Il n'y a pas dans les circuits ECL production de pointes parasites durant les transitions d'états.
   e) Les dispositifs ECL consomment moins que les circuits TTL standard.
   f) Il est facile d'associer un ECL à un TTL.

8-10 CIRCUITS INTÉGRÉS NUMÉRIQUES MOS

La technologie MOS (métal-oxyde-semiconducteur) tire son nom de la façon dont sont constitués les éléments: une électrode métallique sur un oxyde qui sert d'isolant entre l'électrode et le substrat semiconducteur. Les transistors de la technologie MOS sont des transistors à effet de champ appelés MOS. La plupart des CI numériques MOS sont réalisés seulement avec des transistors MOS et avec rien d'autre.

Les principaux avantages des transistors MOS sont leur facilité de fabrication, leur faible coût, leur petite dimension et leur faible consommation. Les CI MOS sont presque 30 % plus facile à fabriquer que les CI bipolaires (TTL, ECL etc.). En outre, les dispositifs MOS prennent moins d'espace sur une puce que les transistors BIPOLAIRES. De façon caractéristique, un transistor MOS prend une superficie de 1 mil carré tandis qu'un transistor bipolaire utilise une superficie de 50 mil carrés. D'ailleurs, et c'est encore plus important, les CI numériques MOS n'utilisent habituellement pas d'éléments résistifs intégrés qui prennent tellement de place dans les CI bipolaires.

On peut donc placer sur un CI MOS un plus grand nombre d'éléments de circuit que sur une puce bipolaire de même dimension. C'est d'ailleurs pour cela que les CI MOS surclassent les CI bipolaires dans le domaine de l'intégration à grande échelle (LSI). La grande densité d'intégration des CI MOS confère à ces systèmes une plus grande fiabilité puisqu'il y a moins de connexions externes à faire.

Le principal inconvénient des CI MOS est leur vitesse de fonctionnement relativement faible par rapport à celle des familles bipolaires. Dans les nombreuses applications où cet aspect n'est pas critique, la logique MOS est de loin supérieure à la logique bipolaire. Après une courte description du transistor MOS, nous examinerons les caractéristiques des familles logiques MOS.

8-11 TRANSISTOR MOS

Il existe actuellement deux types de transistors à effet de champ MOS: celui à *appauvrissement* et celui à *enrichissement*. Les circuits intégrés numériques MOS utilisent seulement le transistor MOS à enrichissement, par conséquent nous n'étudierons que ce type de transistor. En outre, notre étude ne portera que sur le fonctionnement du transistor MOS comme interrupteur.

La figure 8-33 schématise les symboles des transistors MOS à enrichissement canal-*N* et canal-*P*; sur ces schémas le sens de la flèche indique soit un canal-*N*, soit un canal-*P*. Il y a sur ces symboles une ligne brisée entre la *source* et le *drain* pour indiquer que *normalement* il n'y a pas de canal conducteur entre ces électrodes. On voit également sur ces symboles une séparation entre la *grille* et les autres bornes afin de bien montrer que la résistance est très élevée (généralement de l'ordre de 10 000 MΩ).

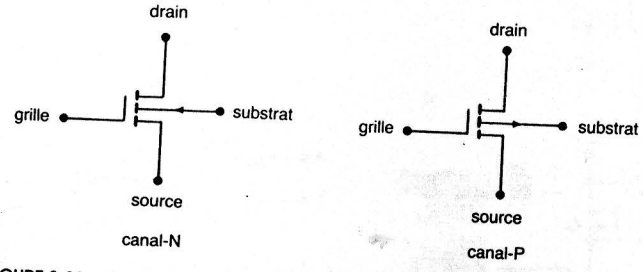


FIGURE 8-33 Symboles schématiques des transistors MOS à enrichissement.

Interrupteur MOS de base Vous pouvez voir à la figure 8-34 comment a lieu la commutation avec un transistor MOS à canal-*N*. Dans un tel dispositif, le drain est toujours polarisé en direct par rapport à la source. Remarquez que le substrat est également connecté à la source, comme c'est souvent le cas. La tension grille-source V_{GS} est la tension d'entrée

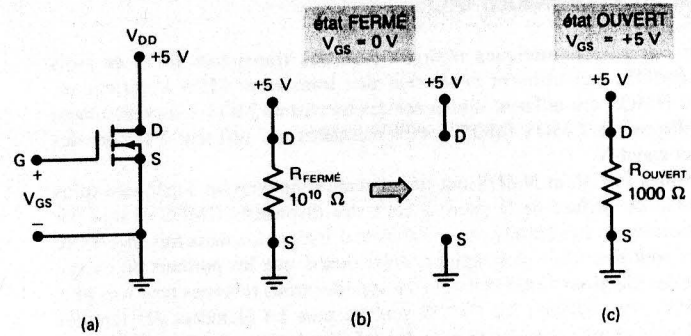


FIGURE 8-34 États ouvert et fermé d'un transistor MOS canal-*N*.

que l'on utilise pour régler la résistance entre le drain et la source (la résistance du canal) et donc pour déterminer si le dispositif est OUVERT ou FERMÉ.

Quand $V_{GS} = 0$ V, il n'y a pas de canal conducteur entre la source et le drain et le dispositif est FERMÉ. La résistance typique du canal dans l'état FERMÉ est $10^{10}\,\Omega$, ce qui, à toutes fins pratiques, correspond à un *circuit ouvert*. Le transistor MOS demeure fermé tant que V_{GS} est 0 ou négatif. Si on applique une tension positive sur la grille par rapport à la source (V_{GS} est positif), on atteint éventuellement une tension seuil (V_S), pour laquelle le canal devient conducteur entre la source et le drain. La valeur caractéristique de V_S est +1,5 V pour un transistor NMOS; le transistor MOS est donc conducteur chaque fois que $V_{GS} \geq 1{,}5$ V. Habituellement, la valeur V_{GS} ouvrant complètement le transistor MOS est plus grande que V_S. Comme on peut le voir à la figure 8-34 b), quand $V_{GS} = +5$ V, la résistance du canal entre la source et le drain diminue jusqu'à $R_{OUVERT} = 1000\,\Omega$.

On résume au tableau 8-6 les caractéristiques de commutation des dispositifs canal-*P* et canal-*N*.

Fondamentalement, un transistor NMOS passe d'une très haute résistance à une très basse résistance quand la tension de la grille passe d'un niveau BAS à un niveau HAUT. Il est commode de s'imaginer que le transistor MOS est un interrupteur qui est soit ouvert, soit fermé entre la source et le drain.

Le transistor MOS canal-*P* fonctionne exactement de la même façon que le dispositif à canal-*N*, à l'exception de la polarité des tensions qui est inversée. Dans le cas du transistor PMOS, le drain est raccordé à une tension d'alimentation V_{DD} négative par rapport à la source. Pour ouvrir le transistor PMOS, on doit appliquer à la grille une tension négative qui dépasse V_S.

TABLEAU 8-6

	Polarisation drain-source	Tension grille-source (V_{GS}) assurant la conduction	R_{OUVERT}	$R_{FERMÉ}$
canal-*P*	négatif	négative et plus négative que V_S	1000 Ω valeur type	10^{10} Ω
canal-*N*	positif	positive et supérieure à V_S	1000 Ω valeur type	10^{10} Ω

8-12 CIRCUITS NUMÉRIQUES MOS

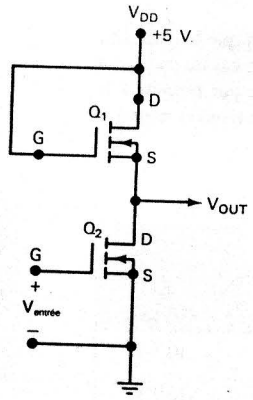
On peut classer les circuits numériques réalisés avec des transistors MOS en trois catégories: 1) les PMOS, qui utilisent *seulement* des transistors MOS à enrichissement canal-*P*; 2) les NMOS qui utilisent *seulement* des transistors MOS à enrichissement canal-*N*; et 3) les dispositifs CMOS (MOS complémentaire) qui utilisent à la fois des dispositifs canal-*P* et canal-*N*.

Les CI numériques PMOS et NMOS ont une densité d'intégration supérieure (plus de transistors par unité de surface de la puce) à celle des dispositifs CMOS et sont par conséquent plus économiques. Les NMOS ont une densité d'intégration deux fois supérieure à celle des PMOS et sont deux fois plus rapides, étant donné que les porteurs de charge dans les NMOS sont des électrons libres alors que ce sont des trous (charges positives plus lentes) dans les PMOS. Par ailleurs, les CMOS sont de tous les éléments des familles MOS ceux qui ont la plus grande complexité et la densité d'intégration la plus faible, mais ce sont également ceux qui sont les plus rapides et qui consomment le moins.

Dans la présente section, nous allons étudier quelques circuits logiques NMOS de base; tous les circuits que nous étudierons ont leur équivalent en PMOS, la seule différence étant la polarité des tensions. Étant donné que les éléments PMOS et NMOS sont utilisés surtout dans des dispositifs LSI et VLSI (microprocesseurs, mémoires, mémoires mortes, etc.), nous réservons à un autre chapitre l'étude des applications utilisant ces familles. La famille CMOS se retrouve surtout dans les applications MSI, comme la famille TTL, et nous l'étudierons plus en détail à partir de la section 8-14.

INVERSEUR NMOS La figure 8-35 reproduit le circuit INVERSEUR NMOS de base. On y trouve deux transistors MOS canal-*N*: Q_1 appelé le transistor MOS de *charge* et Q_2 le transistor MOS de *commutation*. La grille de Q_1 est *toujours* connectée à +5 V, de sorte que ce transistor est toujours OUVERT et se comporte essentiellement comme une résistance de charge de valeur R_{OUVERT}. Q_2 s'ouvre ou se ferme en réponse à la tension $V_{ENTRÉE}$. Le transistor MOS Q_1 est conçu pour avoir un canal plus étroit que Q_2, de sorte que la résistance R_{OUVERT} de Q_1 est supérieure à celle de Q_2. Les valeurs caractéristiques de

FIGURE 8-35 INVERSEUR NMOS.



a)

$V_{ENTRÉE}$	Q_1	Q_2	$V_{SORTIE} = \overline{V_{ENTRÉE}}$
0 V 0 logique	R_{OUVERT} = 100 kΩ	$R_{FERMÉ}$ = 10^{10} Ω	+5 V 1 logique
+5 V 1 logique	R_{OUVERT} = 100 kΩ	R_{OUVERT} = 1 kΩ	+0,05 V 0 logique

b)

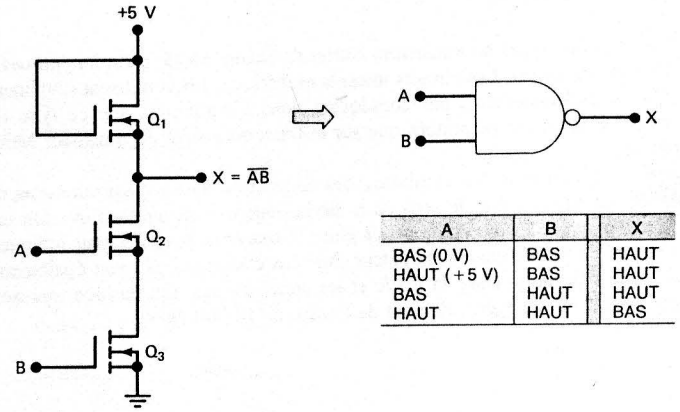
R_{OUVERT} sont pour Q_1 100 kΩ et pour Q_2, 1 kΩ. $R_{FERMÉ}$ pour Q_2 est généralement de l'ordre de 10^{10} Ω.

À la figure 8-35 b), on a résumé sous forme de tableau les deux états de l'INVERSEUR. La meilleure façon d'analyser ce circuit est de supposer que chaque canal d'un transistor MOS est une résistance et que la tension de sortie est prélevée au milieu d'un diviseur de tension formé par ces deux résistances. Quand $V_{ENTRÉE}$ = 0 V, le transistor Q_2 est FERMÉ et sa résistance est très grande (environ 10^{10} Ω). Comme Q_1 a R_{OUVERT} = 100 kΩ, la sortie du diviseur de tension est pratiquement +5 V. Si $V_{ENTRÉE}$ = +5 V, Q_2 est OUVERT et R_{OUVERT} = 1 kΩ. Les résistances du diviseur de tension sont maintenant 100 kΩ et 1 kΩ, de sorte que V_{SORTIE} = 1/101 × (+5 V) ≈ 0,05 V.

Ce circuit réalise donc la fonction d'un INVERSEUR, puisque quand l'entrée est BASSE la sortie est HAUTE et vice versa. Cet INVERSEUR de base peut être modifié pour obtenir des portes logiques NON-ET et NI.

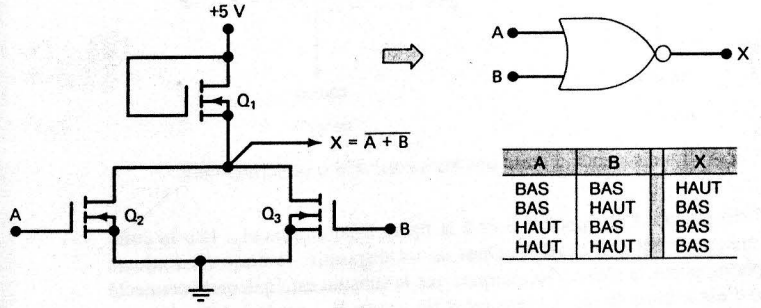
Porte NON-ET NMOS L'opération NON-ET est effectuée par le circuit de la figure 8-36 a) sur laquelle Q_1 joue de nouveau le rôle d'une résistance de charge tandis que Q_2 et

FIGURE 8-36 a) Porte NON-ET NMOS; b) porte NI.



A	B	X
BAS (0 V)	BAS	HAUT
HAUT (+5 V)	BAS	HAUT
BAS	HAUT	HAUT
HAUT	HAUT	BAS

a)



A	B	X
BAS	BAS	HAUT
BAS	HAUT	BAS
HAUT	BAS	BAS
HAUT	HAUT	BAS

b)

Q_3 sont des interrupteurs commandés par les niveaux d'entrée A et B. Si A ou B est à 0 V (0 logique), le transistor à effet de champ correspondant est FERMÉ, de sorte que la résistance entre la borne de sortie et la masse est très élevée et que la tension sur X est HAUTE (+5 V). Quand la tension sur A et B est +5 V (1 logique), les deux transistors Q_2 et Q_3 sont OUVERTS et la sortie X est BASSE. On se convainc donc sans difficulté que la tension de la sortie est égale à la mise en fonction NON-ET des entrées ($X = \overline{AB}$).

Porte NI NMOS La porte NI de la figure 8-36 b) est formée des deux transitors Q_2 et Q_3 qui fonctionnent comme des interrupteurs mis en parallèle, et de Q_1 qui, de la même façon que ci-dessus, agit comme une résistance de charge. Quand l'une des entrées, A ou B, est à +5 V, le transistor MOS correspondant est OUVERT, ce qui amène la sortie au niveau BAS. Quand les deux entrées sont à 0 V, Q_2 et Q_3 sont tous les deux FERMÉS, de sorte que la sortie se trouve au niveau HAUT. Ici aussi, on se convainc facilement que l'expression de sortie est celle de la mise en fonction NI des entrées et que $X = \overline{A + B}$.

Bascules NMOS On fabrique des bascules en rétrocouplant deux portes NI ou NON-ET. La bascule MOS est un élément très important des mémoires MOS et nous en reparlerons plus loin.

8-13 CARACTÉRISTIQUES DE LA LOGIQUE MOS

Les familles logiques MOS comparées aux familles logiques bipolaires sont plus lentes, consomment moins d'énergie, ont des marges de sensibilité aux bruits plus larges, fonctionnent sous une gamme de tensions d'alimentation plus grande, ont une sortance plus élevée et, comme on l'a déjà dit, ont une densité d'intégration supérieure.

Vitesse de fonctionnement Le retard de propagation caractéristique d'une porte NON-ET NMOS est de 50 ns. Deux facteurs expliquent ce retard: la résistance de sortie relativement élevée (100 kΩ) dans l'état HAUT et la charge capacitive des bornes d'entrée des circuits logiques pilotés. Les entrées logiques MOS ont des résistances d'entrée très élevées ($> 10^{12}$ Ω) et ont un condensateur MOS assez élevé, entre 2 et 5 picofarads. C'est cette combinaison d'une R_{SORTIE} élevée et d'un C_{CHARGE} élevé qui allonge le temps de commutation.

Marge de sensibilité aux bruits Ces marges pour les dispositifs NMOS sont d'environ 1,5 V quand la tension d'alimentation $V_{DD} = 5$ V et augmentent proportionnellement à mesure qu'on alimente le drain avec des tensions plus élevées.

Sortance La résistance d'entrée extrêmement élevée des entrées à transistor MOS explique le fait que la sortance des familles logiques MOS est presque illimitée. D'ailleurs, la sortance est vraiment illimitée pour le fonctionnement en courant continu et à basse fréquence. Toutefois, pour des fréquences supérieures à 100 kHz, les capacités d'entrée des grilles affectent le temps de commutation qui augmente proportionnellement aux nombres de charges excitées. Même dans ce cas, la logique MOS fonctionne sans problème avec une sortance de 50, ce qui est déjà beaucoup plus que les dispositifs des familles bipolaires.

Consommation Les circuits logiques MOS consomment peu d'énergie parce que leurs résistances sont relativement grandes. Voyons un exemple de ceci; on peut calculer la dissipation d'énergie de l'INVERSEUR de la figure 8-35 pour ses deux états de fonctionnement.

1. $V_{ENTRÉE} = 0$ V: $R_{OUVERT(Q_1)} = 100$ kΩ; $R_{FERMÉ(Q_2)} = 10^{10}$ Ω. Donc le courant I_D de l'alimentation $V_{DD} \approx 0{,}05$ nA, et $P_D = 5$ V × 0,05 nA = 0,25 nW.
2. $V_{ENTRÉE} = +5$ V: $R_{OUVERT(Q_1)} = 100$ kΩ; $R_{OUVERT(Q_2)} = 1$ kΩ. Donc, $I_D = 5$ V/101 kΩ ≈ 50 μA et $P_D = 5$ V × 50 μA = 0,25 mW.

On a donc un P_D *moyen* un peu supérieur à 0,1 mW pour l'INVERSEUR. Cette faible consommation de la logique MOS la destine tout particulièrement aux applications LSI et VLSI dans lesquelles on peut placer de nombreuses portes et bascules sans risque de surchauffer la puce.

Complexité des applications La famille logique MOS est la famille logique la plus simple à fabriquer puisqu'elle ne contient qu'un élément de base: le transistor NMOS (ou PMOS). Elle n'a besoin d'aucun autre élément, comme des diodes, des résistances etc. Cette caractéristique, de même que la faible consommation P_D, en font l'élément idéal pour les applications LSI (mémoires de grande capacité, puces de calculatrice, etc.) et c'est la raison pour laquelle la logique MOS a fait faire un important bond en avant aux techniques numériques. La vitesse de fonctionnement des PMOS et NMOS ne se compare pas à celle des TTL, et c'est pourquoi on retrouve peu des éléments de la famille MOS dans les applications SSI et MSI. En fait, on ne retrouve presque aucun circuit MOS dans la catégorie des dispositifs SSI ou MSI (portes, bascules, compteurs etc.). Par contre, la famille CMOS a réussi à percer dans le domaine des circuits MSI qui était auparavant dominé par la famille TTL.

Sensibilité à l'électricité statique À des degrés divers, tous les dispositifs électroniques manifestent une certaine sensibilité aux charges électriques statiques. Le corps humain est un vaste réservoir de charges électrostatiques. Par exemple, une personne qui marche sur un tapis peut accumuler dans son organisme suffisamment de charges statiques pour être à un potentiel de 30 000 V. Si cette personne touche un dispositif électronique, une partie de cette charge non négligeable est communiquée à l'élément. Si la charge est trop importante, le dispositif peut être détruit.

La famille MOS est tout particulièrement vulnérable à une destruction d'origine électrostatique, tandis que les familles logiques basées sur le transistor bipolaire le sont beaucoup moins. Cela s'explique directement par la haute impédance présente à l'entrée d'un transistor MOS. Une petite charge qui traverse cette haute impédance peut donner lieu à une forte tension disruptive. La même charge dans l'impédance relativement faible d'une entrée TTL ou ECL cause une tension plus petite et moins dommageable.

Le phénomène de la **décharge électrostatique (DES)** est responsable, chaque année, de milliards de dollars de dommages à l'équipement électronique, et les fabricants de matériel ont étudié attentivement des solutions spéciales de manipulation des circuits et des dispositifs électroniques. Voici les précautions les plus courantes que l'on prend dans les laboratoires de génie, les usines de fabrication et les services de réparation:

1. Connecter le châssis de tous les appareils d'essai, pointes de fer à souder et bancs d'essai (en métal) à une masse à la terre (par ex. la broche cylindrique d'une fiche d'alimentation 120 V). Cela empêche l'accumulation de charges

statiques sur ces éléments qui pourraient être transférées aux plaquettes ou aux CI au moment d'un contact.

2. Se connecter à la terre au moyen d'un bracelet spécial. Cela a pour effet de retourner à la terre les dangereuses charges qui peuvent s'accumuler dans notre organisme. Ce bracelet antistatique renferme une résistance de 1 MΩ qui limite le courant à une intensité non léthale en cas d'électrocution pendant le travail.
3. Enfoncer les CI (tout particulièrement les MOS) dans une mousse conductrice ou un papier d'aluminium. Cela court-circuite toutes les broches et prévient le développement d'une différence de potentiel dangereuse entre deux broches.
4. Emballer les plaquettes de circuits intégrés dans des enveloppes plastiques ou métalliques.
5. Ne jamais laisser des entrées de CI non connectées, car les entrées ouvertes ont tendance à capter des charges statiques parasites.

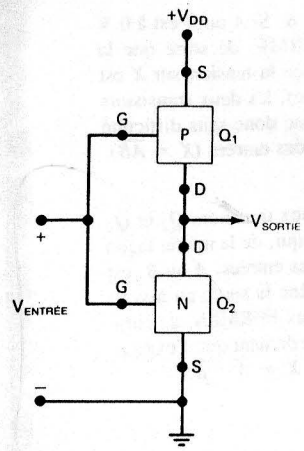
QUESTIONS DE RÉVISION

1. Indiquez les avantages et les inconvénients relatifs aux circuits NMOS par rapport aux circuits TTL.
2. Dites ce qui limite la sortance de la logique MOS.
3. Indiquez les facteurs limitant la vitesse de commutation des circuits NMOS.
4. Dites pourquoi les circuits MOS conviennent si bien à l'intégration à grande échelle.
5. Pourquoi les CI MOS sont-ils plus sensibles aux charges électrostatiques?
6. Énumérez les précautions à prendre lors de la manipulation des CI et des cartes imprimées pour se prémunir contre l'électrostatique.

8-14 LOGIQUE MOS COMPLÉMENTAIRE

La famille logique *MOS complémentaire* utilise les *deux* types de transistor MOS à canal-*N* et à canal-*P* dans le même circuit en vue de retirer des avantages que ne possèdent pas les familles PMOS et NMOS. À proprement parler, on peut dire que les CMOS sont plus rapides et encore moins gourmands que les autres familles MOS. Ces avantages sont quelque peu atténués par la complexité de fabrication plus grande de ces CI et par leur densité d'intégration moins élevée. Donc les CMOS ne peuvent prétendre compétitionner les familles MOS dans le cas des applications où l'intégration de dispositifs doit être la plus poussée possible.

Toutefois, la logique CMOS s'impose de plus en plus dans le secteur des dispositifs MSI, principalement aux dépens de la famille TTL, qu'elle concurrence directement. Le procédé de fabrication des CMOS est plus simple que celui des TTL et sa densité d'intégration est plus élevée, ce qui permet donc de mettre plus de circuits par unité de surface et de réduire le coût par fonction. Les CMOS ne consomment seulement qu'une fraction de l'énergie dissipée par la série TTL faible consommation (74L) et conviennent donc parfaitement aux applications fonctionnant sur pile ou dont l'alimentation de secours est sur pile. Normalement, la vitesse de fonctionnement des CMOS est inférieure au TTL, mais la nouvelle série CMOS rapide concurrence désormais les séries 74 et 74LS.



$V_{ENTRÉE}$	Q_1	Q_2	V_{SORTIE}
$+V_{DD}$ 1 logique	SORTIE $R_{SORTIE} = 10^{10}\ \Omega$	ENTRÉE $R_{ENTRÉE} = 1\ k\Omega$	$\simeq 0$ V
0 V 0 logique	ENTRÉE $R_{ENTRÉE} = 1\ k\Omega$	SORTIE $R_{SORTIE} = 10^{10}\ \Omega$	$\simeq +V_{DD}$

$V_{SORTIE} = \overline{V_{ENTRÉE}}$

FIGURE 8-37 INVERSEUR CMOS de base.

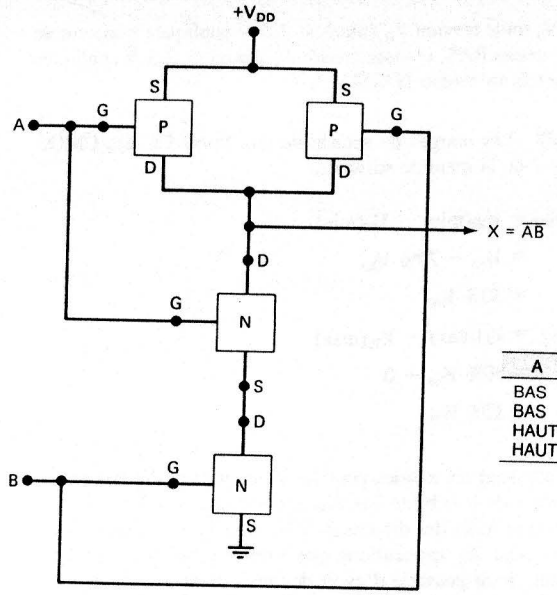
Inverseur CMOS Le circuit de l'INVERSEUR CMOS de base est reproduit à la figure 8-37. Sur ce schéma et sur ceux qui suivent, on a remplacé le symbole courant du transistor MOS par des blocs portant la lettre *P* et *N* pour indiquer respectivement un transistor MOS-P et un transistor MOS-N. On procède ainsi tout simplement parce que l'analyse des circuits est plus commode. Le montage de l'INVERSEUR CMOS est constitué de deux transistor MOS en série; dans ce montage le dispositif canal-*P* a sa source connectée à $+V_{DD}$ (tension positive) et le dispositif canal-*N* a sa source connectée à la masse*. Les grilles des deux dispositifs sont réunies pour former une entrée commune. Les drains de ces deux dispositifs sont aussi réunis pour former une sortie commune.

Les niveaux logiques d'un dispositif CMOS sont essentiellement $+V_{DD}$ pour le 1 logique et 0 V pour le 0 logique. Considérons d'abord le cas où $V_{ENTRÉE} = +V_{DD}$. Dans cette situation, la grille de Q_1 (canal-*P*) est à 0 V par rapport à la source de Q_1. Ainsi Q_1 est-il FERMÉ et sa $R_{FERMÉ} \approx 10^{10}\ \Omega$. La grille de Q_2 (canal-*N*) est alors à la tension $+V_{DD}$ par rapport à sa source. Donc ce transistor est OUVERT et $R_{OUVERT} = 1\ k\Omega$. Le diviseur de tension entre $R_{FERMÉ}$ de Q_1 et R_{OUVERT} de Q_2 donne lieu à un $V_{SORTIE} \approx 0$ V.

Dans un deuxième cas, considérons que $V_{ENTRÉE} = 0$ V. La grille de Q_1 est maintenant à un potentiel négatif par rapport à sa source tandis que V_{GS} de $Q_2 = 0$ V. C'est donc Q_1 qui est OUVERT avec $R_{OUVERT} = 1\ k\Omega$ et Q_2 qui est FERMÉ avec $R_{FERMÉ} = 10^{10}\ \Omega$. La tension V_{SORTIE} est maintenant environ $+V_{DD}$. Le tableau de la figure 8-37 b) résume le fonctionnement dans ces deux états; on voit que ce circuit se comporte comme un INVERSEUR logique.

Porte NON-ET CMOS En modifiant l'INVERSEUR de base on parvient à réaliser d'autres fonctions logiques. Par exemple, la figure 8-38 schématise une porte NON-ET réalisée en

* La plupart des fabricants appellent la borne de la masse V_{SS}.



A	B	X
BAS	BAS	HAUT
BAS	HAUT	HAUT
HAUT	BAS	HAUT
HAUT	HAUT	BAS

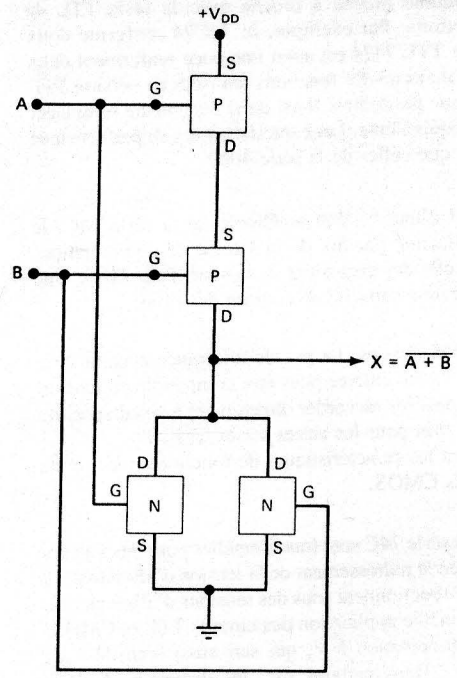
FIGURE 8-38 Porte NON-ET CMOS.

ajoutant à l'inverseur de base un transistor MOS canal-*P* en parallèle et un transistor MOS canal-*N* en série. L'analyse de ce genre de circuit se fait plus facilement si on réalise qu'une entrée de 0 V OUVRE le transistor MOS-P correspondant et FERME le transistor MOS-N correspondant et qu'une entrée de $+V_{DD}$ a l'effet tout à fait contraire. Il est donc facile de constater que la seule fois que la sortie est au niveau BAS survient lorsque les entrées *A* et *B* sont toutes les deux au niveau HAUT ($+V_{DD}$). Les deux transistors MOS-N sont alors ouverts, ce qui place une faible résistance entre la borne de sortie et la masse. Dans toutes les autres conditions, au moins un transistor MOS-P est OUVERT tandis qu'au moins un transistor MOS-N est FERMÉ, ce qui donne en sortie un niveau HAUT.

Porte NI CMOS Une porte NI CMOS s'obtient en ajoutant à l'inverseur de base en série un transistor MOS-P et en parallèle un transistor MOS-N, voir figure 8-39. De nouveau, on analyse ce circuit en se rendant compte que toute entrée au niveau BAS OUVRE son transistor MOS-P correspondant et FERME son transistor MOS-N correspondant et vice versa dans le cas d'une entrée au niveau HAUT. Nous laissons le soin au lecteur de vérifier que ce circuit fonctionne comme une porte NI.

Pour obtenir des portes OU et ET CMOS, il suffit d'ajouter des inverseurs aux portes NON-ET et NI.

Bascules S-C CMOS Si on rétrocouple deux portes NON-ET ou deux portes NI CMOS, on obtient une bascule S-C ordinaire. Pour fabriquer des bascules synchrones ou J-K à partir de la bascule S-C de base, on doit ajouter d'autres portes logiques.



A	B	X
BAS	BAS	HAUT
BAS	HAUT	BAS
HAUT	BAS	BAS
HAUT	HAUT	BAS

FIGURE 8-39 Porte NI CMOS.

QUESTIONS DE RÉVISION

1. *Vrai ou faux:*
   a) Le CMOS est le plus rapide mais aussi le plus gourmand en énergie de tous les membres de la famille MOS.
   b) En technologie CMOS, on peut intégrer plus de circuits sur une puce qu'en technologie TTL.
   c) Dans un INVERSEUR CMOS, les deux transistors MOS conduisent en même temps.
   d) Les dispositifs CMOS conviennent parfaitement aux circuits à piles.

8-15 CARACTÉRISTIQUES DES SÉRIES CMOS

Série 4000/14000 On retrouve un grand nombre de séries différentes dans la famille de circuits numériques CMOS. La première série CMOS à voir le jour est la série 4000 produite par RCA (14000 par Motorola Inc.). La série 4000 originale est la série 4000; la série 4000B est une variante améliorée fournissant un plus fort courant de sortie. Ces deux séries sont encore très usitées en dépit de l'apparition de nouvelles séries CMOS. La série 4000 subsiste depuis assez longtemps, malgré cela elle possède de nombreuses fonctions encore introuvables dans les nouvelles versions.

Série 74C Cette série CMOS est compatible broche à broche avec la série TTL de même numéro et réalise les mêmes fonctions. Par exemple, le 74C74 renferme deux bascules D déclenchées par un signal, et le TTL 7474 est aussi une puce renfermant deux bascules D déclenchées par un signal. Une majorité des fonctions réalisées en version TTL se retrouvent dans la version CMOS, quoique pas toutes. Il est donc faisable de remplacer des circuits TTL par une réalisation CMOS équivalente. Les caractéristiques de performance de la série 74C sont à peu près les mêmes que celles de la série 4000.

Série 74HC (CMOS rapides) Il s'agit ici d'une version améliorée de la série 74C, le perfectionnement majeur étant une multiplication par dix de la vitesse de commutation. La vitesse de ces dispositifs se compare à celle des dispositifs de la série TTL 74LS. Une autre amélioration à souligner est la plus grande capacité de courant de sortie.

Série 74HCT Voici une autre série de CMOS rapides. La grande différence avec la série précédente est que cette dernière (la 74HCT) a été étudiée pour être compatible en tension avec les dispositifs TTL. Autrement dit, on peut les raccorder directement à des dispositifs TTL. Comme nous le verrons, ce n'est pas vrai pour les autres séries CMOS.

Les paragraphes suivants vous exposent les caractéristiques de fonctionnement et de performance communes à tous les dispositifs CMOS.

Tension d'alimentation La série 4000 et la série 74C sont fonctionnelles pour des valeurs de V_{DD} comprises entre 3 et 15 V, de sorte que le redressement de la tension d'alimentation n'est pas critique. Les séries 74HC et 74HCT fonctionnent sous des tensions d'alimentation allant de 2 à 6 V. Quand on utilise dans une même application des circuits TTL et CMOS, on retient généralement comme tension d'alimentation 5 V, qui sert aussi bien de V_{DD} pour les CMOS que de V_{CC} pour les TTL. Dans certains cas, les dispositifs CMOS fonctionneront à des tensions supérieures à 5 V et il faudra alors prévoir des solutions pour permettre aux dispositifs CMOS et TTL de fonctionner ensemble. Nous examinerons plus loin ce genre de solutions.

TABLEAU 8-7 NIVEAUX DE TENSION CMOS (4000B)

$V_{OL}(\text{max})$	= 0 V
$V_{OH}(\text{min})$	= V_{DD}
$V_{IL}(\text{max})$	= 30 % de V_{DD}
$V_{IH}(\text{min})$	= 70 % de V_{DD}

Niveaux de tension Dans un circuit tout CMOS, les niveaux de tension d'un CI CMOS sont très près de 0 V pour le niveau BAS et de V_{DD} pour le niveau HAUT. Il en est ainsi du fait que la résistance d'entrée d'un CMOS est tellement élevée (10^{12} ohms) qu'elle n'impose pas de charge à la sortie du CMOS qui la pilote. Les exigences des tensions d'entrée pour les deux états logiques sont exprimées sous forme d'un pourcentage de la tension d'alimentation V_{DD}, $V_{IL}(\text{max})$ est fixée à 30 % de V_{DD} et $V_{IH}(\text{min})$, à 70 % de V_{DD}.* Ces niveaux de tension sont illustrés au tableau 8-7.

* Pourcentage valables pour la série 4000B. Vous trouverez les valeurs V_{IH} et V_{IL} des séries 74HC et 74HCT à la section 8-18.

Par exemple, si V_{DD} = 5 V, toute tension $V_{IL}(\text{max})$ = 1,5 V appliquée à une entrée CMOS sera pour le dispositif un niveau BAS, et toute tension $V_{IH}(\text{min})$ = 3,5 V appliquée à une entrée sera considérée comme un niveau HAUT.

Marges de sensibilité aux bruits Les marges de sensibilité aux bruits CC des CMOS se calculent à partir du tableau 8-7 de la manière suivante:

$$\begin{aligned} V_{NH} &= V_{OH}(\text{min}) - V_{IH}(\text{min}) \\ &= V_{DD} - 70\%\ V_{DD} \\ &= 30\%\ V_{DD} \\ V_{NL} &= V_{IL}(\text{max}) - V_{OL}(\text{max}) \\ &= 30\%\ V_{DD} - 0 \\ &= 30\%\ V_{DD} \end{aligned}$$

Les marges de sensibilité aux bruits sont les mêmes pour les deux états et dépendent de V_{DD}. Quand V_{DD} = 5 V, les marges de sensibilité aux bruits sont toutes les deux 1,5 V. Elles sont sensiblement meilleures que celles des dispositifs TTL et ECL. Les CMOS sont donc particulièrement intéressants pour les applications que l'on doit monter dans des milieux très parasités. Évidemment, il est possible d'avoir des marges de sensibilité aux bruits encore meilleures en utilisant une tension V_{DD} plus grande. Cette amélioration de l'immunité aux bruits se fait toutefois au détriment de la consommation à cause de l'alimentation plus élevée.

Consommation Quand un circuit logique CMOS est dans un état statique (ne change pas), sa consommation est extrêmement faible. La raison de ceci est donnée par les circuits des figures 8-37 à 8-39. Notez que quel que soit l'état de la sortie, il y a toujours une résistance très élevée entre la borne V_{DD} et la masse parce qu'il y a toujours un transistor MOS de FERMÉ dans la trajectoire du courant. La consommation d'un CMOS typique n'est donc que de 2,5 nW par porte quand V_{DD} = 5 V; cette consommation ne dépasse pas 10 nW quand V_{DD} = 10 V. Avec de telles valeurs pour P_D, on conçoit sans difficulté que les CMOS soient les circuits idéaux pour les applications où la consommation est un élément essentiel.

P_D augmente avec la fréquence La consommation d'un CI CMOS demeure faible tant que nous sommes en régime continu. Malheureusement, P_D augmente proportionnellement à la fréquence de commutation des circuits. Par exemple, une porte NON-ET CMOS dont P_D est 10 nW en régime continu se retrouve avec un P_D de 0,1 mW à la fréquence de 100 kHz et de 1 mW à 1 MHz. La figure 8-40 illustre pourquoi la consommation augmente avec la fréquence.

Chaque fois qu'une sortie de CMOS passe du niveau BAS au niveau HAUT, un courant de charge transitoire est fourni au condensateur de charge. La capacité de celui-ci est la somme des capacités d'entrée de toutes les charges excitées et de la capacité de sortie du dispositif lui-même. Ces pointes étroites de courant sont fournies par V_{DD} et elles peuvent avoir fréquemment une amplitude de 5 mA et une durée de 20 à 30 ns. On conçoit donc facilement qu'à mesure que la fréquence de commutation augmente, il y ait un plus grand nombre de pointes de courant de ce genre par seconde et que le courant moyen prélevé de l'alimentation augmente.

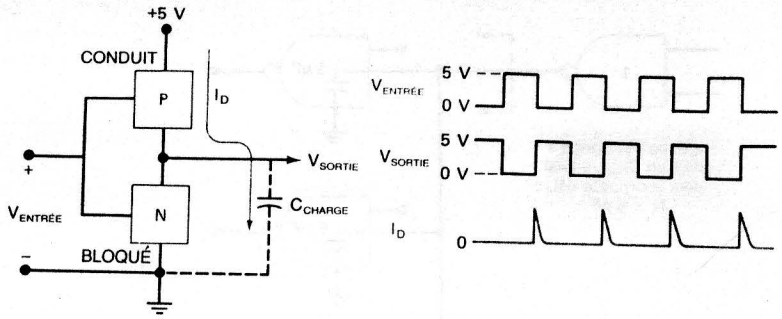


FIGURE 8-40 Pointes de courant extraites de l'alimentation V_{DD} chaque fois que la sortie passe du niveau BAS au niveau HAUT. Ces pointes sont le courant de charge du condensateur de charge.

Ainsi les éléments CMOS perdent-ils, aux hautes fréquences, certains de leurs avantages par rapport aux autres familles logiques. À titre de règle générale, on peut dire qu'une porte CMOS consomme à peu près la même énergie qu'une porte 74LS aux fréquences voisines de 2 à 3 MHz. Dans le cas des puces MSI, la situation est un peu plus complexe que ce que l'on vient de dire, et un concepteur devra donc effectuer une analyse détaillée afin de déterminer si le CMOS consomme moins à une certaine fréquence de fonctionnement.

Sortance Les entrées CMOS, comme les entrées NMOS et PMOS, ont une résistance très grande (10^{12} ohms) qui ne tire pratiquement aucun courant de la source du signal. Toutefois, chaque entrée CMOS a une charge d'environ 5 pF par rapport à la masse. Cette capacité d'entrée limite le nombre d'entrées CMOS qu'une sortie CMOS peut alimenter (voir figure 8-41). La sortie CMOS doit charger et décharger tous les condensateurs d'entrée montés en parallèle, de sorte que le temps de commutation de la sortie augmente proportionnellement avec le nombre de charges alimentées. De façon générale, on peut dire que chaque charge CMOS augmente le retard de propagation du circuit pilote de 3 ns. Par exemple, la porte NON-ET 1 de la figure 8-41 peut avoir un t_{PLH} de 25 ns si elle ne pilote aucune charge; si elle en pilote *vingt*, son retard de propagation pourra aller jusqu'à 25 ns + 20(3 ns) = 85 ns.

Donc la sortance d'un CMOS est fonction du retard de propagation maximal admissible. Habituellement, les sorties CMOS sont limitées à une sortance de 50 dans les régimes de fonctionnement basse fréquence ($\leq$ 1 MHz). Évidemment, la sortance d'un CMOS utilisé en régime haute fréquence sera plus petite.

Vitesse de commutation Bien que les CMOS, comme les NMOS et les PMOS, aient à alimenter des capacités de charge relativement grandes, leur vitesse de commutation est toutefois plus élevée à cause de leur faible résistance de sortie dans une état comme dans l'autre. Rappelons qu'une sortie NMOS doit charger le condensateur de charge à travers une résistance relativement grande (100 kΩ). Par contre, dans un circuit CMOS, la résistance de sortie à l'état HAUT est R_{OUVERT} d'un transistor MOS-P, dont la valeur est habituellement de 1 kΩ ou moins. Grâce à cette petite résistance, la charge des condensateurs est plus rapide.

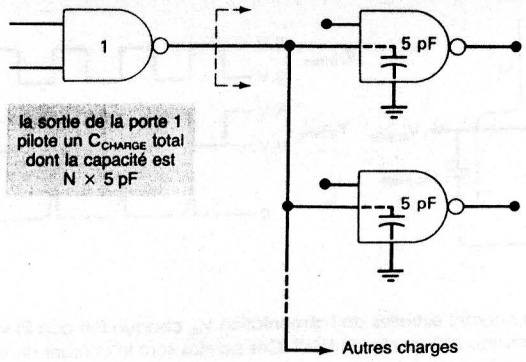


FIGURE 8-41 Chaque entrée CMOS a pour effet d'augmenter la capacité de charge totale imposée à la sortie de la porte pilote.

Une porte NON-ET de la série 4000 a habituellement un t_{pd} moyen de 50 ns à V_{DD} = 5 V, et de 25 ns à V_{DD} = 10 V. Le retard de propagation diminue quand V_{DD} augmente parce que R_{OUVERT} du transistor MOS décroît sensiblement quand la tension d'alimentation augmente. À la lumière de ceci, il semblerait donc que V_{DD} doive être le plus élevé possible pour le fonctionnement en haute fréquence. Évidemment, plus V_{DD} est grand plus le circuit consomme de l'énergie.

Une porte NON-ET type des séries 74HC et 74HCT possède un t_{pd} de 8 ns lorsqu'elle fonctionne sous une tension V_{DD} = 5 V. C'est une vitesse qui se compare à celle de la série 74LS.

Entrées inutilisées *Les entrées CMOS ne doivent jamais être laissées non branchées. Toutes les entrées CMOS doivent être raccordées à une tension fixe (0 V ou V_{DD}) ou à une autre entrée.* Cette règle vaut également pour toutes les entrées des portes logiques inutilisées d'un boîtier. Une entrée CMOS non branchée peut capter des signaux parasites et des charges électrostatiques pouvant facilement polariser les transistors MOS canal-*P* et canal-*N* et les mettre dans leur état conducteur; ceci peut donner lieu à une plus grande consommation et possiblement à une surchauffe.

Susceptibilité aux charges électrostatiques La grande résistance d'entrée des entrées CMOS les rend tout particulièrement vulnérables à l'accumulation de charges électrostatiques qui peuvent générer des tensions suffisamment élevées pour percer l'isolant diélectrique entre la porte du transistor MOS et le canal. Aujourd'hui, la plupart des dispositifs de la série CMOS sont protégés contre les détériorations dues à l'électrostatique, au moyen de diodes Zener de protection placées à chaque entrée. Ces diodes ont été conçues pour s'ouvrir (conduire) et pour limiter la valeur de la tension d'entrée bien en déçà de la valeur disruptive. Généralement, les diodes Zener fonctionnent sans problème, sauf les fois où elles ne s'ouvrent pas assez vite pour empêcher la détérioration du CI; pour cette raison, on recommande de continuer à respecter les consignes de manipulation des circuits déjà énoncées.

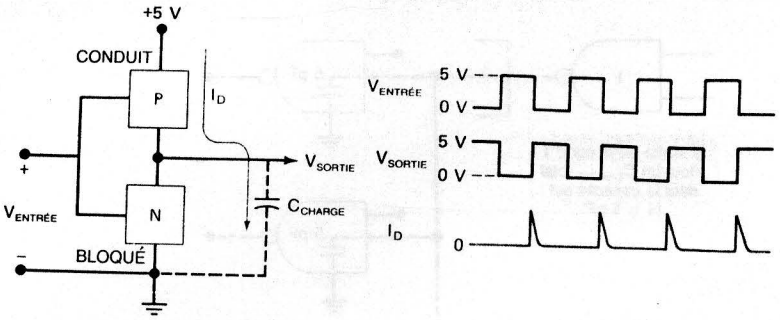


FIGURE 8-40 Pointes de courant extraites de l'alimentation V_{DD} chaque fois que la sortie passe du niveau BAS au niveau HAUT. Ces pointes sont le courant de charge du condensateur de charge.

Ainsi les éléments CMOS perdent-ils, aux hautes fréquences, certains de leurs avantages par rapport aux autres familles logiques. À titre de règle générale, on peut dire qu'une porte CMOS consomme à peu près la même énergie qu'une porte 74LS aux fréquences voisines de 2 à 3 MHz. Dans le cas des puces MSI, la situation est un peu plus complexe que ce que l'on vient de dire, et un concepteur devra donc effectuer une analyse détaillée afin de déterminer si le CMOS consomme moins à une certaine fréquence de fonctionnement.

Sortance Les entrées CMOS, comme les entrées NMOS et PMOS, ont une résistance très grande (10^{12} ohms) qui ne tire pratiquement aucun courant de la source du signal. Toutefois, chaque entrée CMOS a une charge d'environ 5 pF par rapport à la masse. Cette capacité d'entrée limite le nombre d'entrées CMOS qu'une sortie CMOS peut alimenter (voir figure 8-41). La sortie CMOS doit charger et décharger tous les condensateurs d'entrée montés en parallèle, de sorte que le temps de commutation de la sortie augmente proportionnellement avec le nombre de charges alimentées. De façon générale, on peut dire que chaque charge CMOS augmente le retard de propagation du circuit pilote de 3 ns. Par exemple, la porte NON-ET 1 de la figure 8-41 peut avoir un t_{PLH} de 25 ns si elle ne pilote aucune charge; si elle en pilote *vingt*, son retard de propagation pourra aller jusqu'à 25 ns + 20(3 ns) = 85 ns.

Donc la sortance d'un CMOS est fonction du retard de propagation maximal admissible. Habituellement, les sorties CMOS sont limitées à une sortance de 50 dans les régimes de fonctionnement basse fréquence ($\leq$ 1 MHz). Évidemment, la sortance d'un CMOS utilisé en régime haute fréquence sera plus petite.

Vitesse de commutation Bien que les CMOS, comme les NMOS et les PMOS, aient à alimenter des capacités de charge relativement grandes, leur vitesse de commutation est toutefois plus élevée à cause de leur faible résistance de sortie dans une état comme dans l'autre. Rappelons qu'une sortie NMOS doit charger le condensateur de charge à travers une résistance relativement grande (100 kΩ). Par contre, dans un circuit CMOS, la résistance de sortie à l'état HAUT est R_{OUVERT} d'un transistor MOS-P, dont la valeur est habituellement de 1 kΩ ou moins. Grâce à cette petite résistance, la charge des condensateurs est plus rapide.

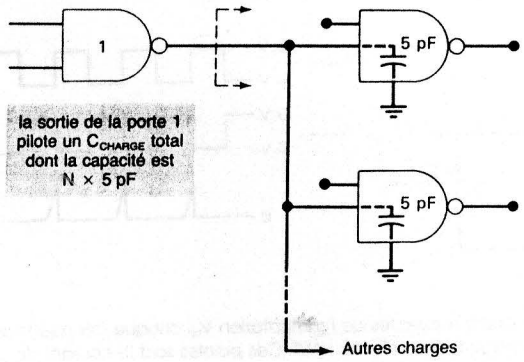


FIGURE 8-41 Chaque entrée CMOS a pour effet d'augmenter la capacité de charge totale imposée à la sortie de la porte pilote.

Une porte NON-ET de la série 4000 a habituellement un t_{pd} moyen de 50 ns à V_{DD} = 5 V, et de 25 ns à V_{DD} = 10 V. Le retard de propagation diminue quand V_{DD} augmente parce que R_{OUVERT} du transistor MOS décroît sensiblement quand la tension d'alimentation augmente. À la lumière de ceci, il semblerait donc que V_{DD} doive être le plus élevé possible pour le fonctionnement en haute fréquence. Évidemment, plus V_{DD} est grand plus le circuit consomme de l'énergie.

Une porte NON-ET type des séries 74HC et 74HCT possède un t_{pd} de 8 ns lorsqu'elle fonctionne sous une tension V_{DD} = 5 V. C'est une vitesse qui se compare à celle de la série 74LS.

Entrées inutilisées *Les entrées CMOS ne doivent jamais être laissées non branchées. Toutes les entrées CMOS doivent être raccordées à une tension fixe (0 V ou V_{DD}) ou à une autre entrée.* Cette règle vaut également pour toutes les entrées des portes logiques inutilisées d'un boîtier. Une entrée CMOS non branchée peut capter des signaux parasites et des charges électrostatiques pouvant facilement polariser les transistors MOS canal-*P* et canal-*N* et les mettre dans leur état conducteur; ceci peut donner lieu à une plus grande consommation et possiblement à une surchauffe.

Susceptibilité aux charges électrostatiques La grande résistance d'entrée des entrées CMOS les rend tout particulièrement vulnérables à l'accumulation de charges électrostatiques qui peuvent générer des tensions suffisamment élevées pour percer l'isolant diélectrique entre la porte du transistor MOS et le canal. Aujourd'hui, la plupart des dispositifs de la série CMOS sont protégés contre les détériorations dues à l'électrostatique, au moyen de diodes Zener de protection placées à chaque entrée. Ces diodes ont été conçues pour s'ouvrir (conduire) et pour limiter la valeur de la tension d'entrée bien en déçà de la valeur disruptive. Généralement, les diodes Zener fonctionnent sans problème, sauf les fois où elles ne s'ouvrent pas assez vite pour empêcher la détérioration du CI; pour cette raison, on recommande de continuer à respecter les consignes de manipulation des circuits déjà énoncées.

TABLEAU 8-8 COMPARAISON DES SÉRIES DE CIRCUITS INTÉGRÉS NUMÉRIQUES*

	74HC	4000B	74	74S	74LS	74AS	74ALS	ECL
Consommation par porte (mW)								
Statique	$2,5 \times 10^{-3}$	$1,0 \times 10^{-3}$	10	20	2	8	1,2	40
À 100 kHz	0,17	0,1	10	20	2	8	1,2	40
Retard de propagation (ns)	8	50	9	3	9,5	1,7	4	1
Vitesse-consommation (à 100 kHz,pJ)	1,4	5	90	60	19	13,6	4,8	40
Fréquence d'horloge maximale (MHz)	40	12	35	12,5	45	200	70	300
Marge aux bruits, cas pessimiste	0,9	1,5	0,4	0,3	0,3	0,3	0,4	0,25

* Toutes ces valeurs CMOS sont valables pour $V_{DD} = 5$ V.

Comparaison des séries CMOS et TTL Le tableau 8-8 compare les caractéristiques types des principales séries de circuits intégrés numériques.

QUESTIONS DE RÉVISION

1. Quelle série CMOS est compatible broche à broche avec la famille TTL?
2. Quels sont les perfectionnements de la série 74HC comparativement à la série 74C?
3. De quelle façon la série 74HCT diffère-t-elle de la série 74HC?
4. Indiquez ce qu'il advient des caractéristiques CMOS suivantes quand V_{DD} augmente: a) la marge de sensibilité aux bruits; b) la consommation; c) la vitesse de commutation.
5. Quels facteurs limitent la sortance des CMOS?
6. Que doit-on faire avec des entrées CMOS non reliées?
7. Quelles précautions doit-on prendre quand on manipule des circuits CMOS?
8. Quelle série de CI convient le mieux aux circuits à piles?
9. Quelle série est la plus appropriée pour un fonctionnement en milieu très parasité?

8-16 CMOS À DRAIN OUVERT ET SORTIES TROIS ÉTATS

Les sorties des CMOS ordinaires ne doivent jamais être connectées ensemble. La figure 8-42 montre ce qui arrive quand les sorties de deux INVERSEURS CMOS sont raccordées ensemble. La sortie de celui de gauche essaie de passer au niveau HAUT, ce qui place en état de conduction son transistor MOS canal-*P* avec $R_{OUVERT} = 1$ kΩ. La sortie de l'autre inverseur tente de se mettre dans l'état BAS, ce qui met dans l'état de conduction son transistor MOS canal-*N* avec $R_{OUVERT} = 1$ kΩ. La borne de sortie commune *X* se retrouve à une tension approximativement égale à $V_{DD}/2$ du fait de l'action du diviseur de tension entre les deux résistances de 1 kΩ.

Cette tension se trouve dans la plage indéterminée (rappelons que V_{IL}(max) = 30 % V_{DD} et V_{IH}(min) = 70 % V_{DD}) et ne peut de ce fait servir à piloter d'autres dispositifs. De plus, le courant qui traverse les deux transistors MOS conducteurs est beaucoup plus intense que le courant normal, particulièrement si V_{DD} est élevé; ceci peut endommager les CI.

Il est relativement facile de savoir si deux sorties CMOS ont été câblées ensemble ou raccordées par inadvertance, puisque les signaux à ces sorties auront trois niveaux

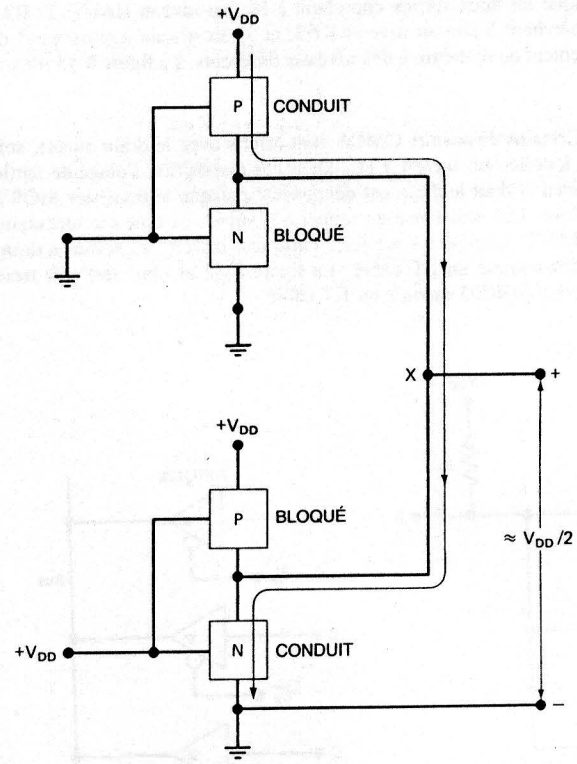
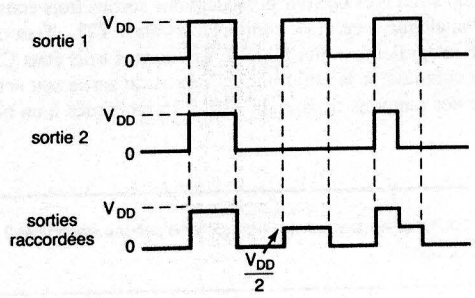


FIGURE 8-42 Quand des sorties sont raccordées, la borne de sortie commune est approximativement à la tension $V_{DD}/2$, si les sorties essaient de se trouver à des niveaux différents.

FIGURE 8-43 Quand deux sorties CMOS sont raccordées, la tension sur la sortie commune égale approximativement $V_{DD}/2$ puisque les deux sorties cherchent à se mettre dans des états différents.



différents: 1) HAUT lorsque les deux sorties cherchent à être au niveau HAUT; 2) BAS quand les deux sorties cherchent à être au niveau BAS; et 3) au niveau approximatif de $V_{DD}/2$ quand les sorties tentent de se mettre à des niveaux différents. La figure 8-43 illustre cette situation.

Sorties à drain ouvert Certains dispositifs CMOS sont offerts avec le drain ouvert, soit la contrepartie des sorties à collecteur ouvert TTL. Dans ces dispositifs, l'étage de sortie est un transistor MOS à canal N dont le drain est déconnecté puisque le transistor MOS à canal P supérieur a été enlevé. Une résistance de rappel à la source externe est nécessaire pour produire un niveau HAUT. Comme les sorties à collecteur ouvert, les sorties à drain ouvert peuvent être montées comme un ET câblé. La figure 8-44 a) nous fait voir trois INVERSEURS à drain ouvert 74HC05 montés en ET câblé.

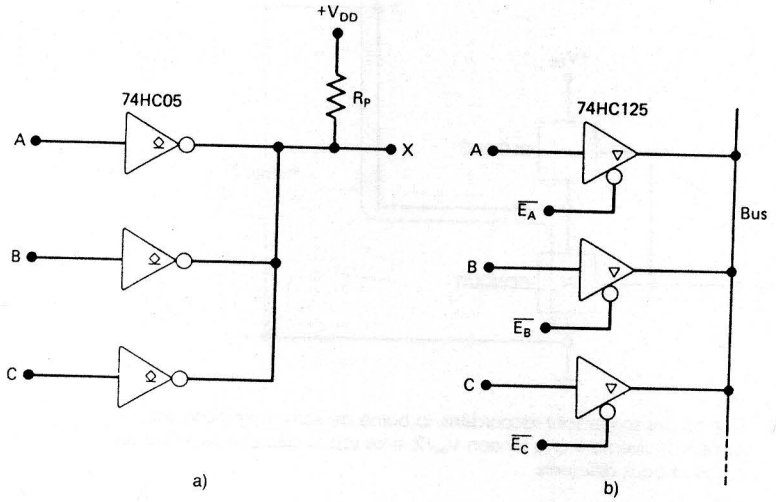


FIGURE 8-44 a) Dispositifs CMOS à drain ouvert réalisant un ET câblé; b) sorties trois états CMOS dans un montage en bus.

Sorties trois états Plusieurs circuits intégrés CMOS possèdent des sorties trois états dont le fonctionnement est tout à fait analogue à celui des sorties trois états TTL. Tout ce qui a été dit pour les TTL s'applique intégralement aux CMOS. Les sorties trois états CMOS peuvent être raccordées à un bus commun, à la condition qu'une seule sortie soit active à la fois. La figure 8-44 b) montre des tampons trois états 74HC125 raccordés à un bus.

QUESTIONS DE RÉVISION

1. Pourquoi les sorties des CMOS courants ne peuvent-elles pas être reliées ensemble?
2. Qu'est-ce qu'une sortie à drain ouvert?

8-17 PORTE DE TRANSMISSION CMOS (INTERRUPTEUR BILATÉRAL)

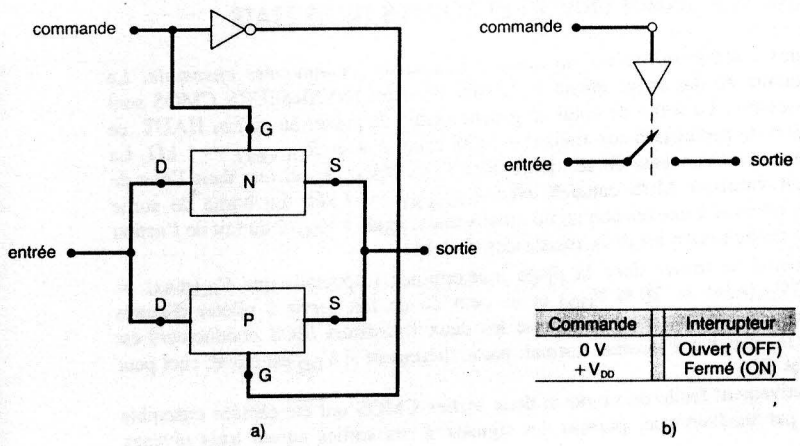
Il existe un circuit CMOS spécial qui n'a pas d'équivalent dans les familles TTL ou ECL; il s'agit de *la porte de transmission* dit *interrupteur bilatéral* qui se comporte essentiellement comme un interrupteur unipolaire unidirectionnel commandé par un signal logique. Cette porte de transmission laisse passer les signaux dans les deux sens et est très utile dans les applications numériques et analogiques.

Le montage de base de l'interrupteur bilatéral est montré à la figure 8-45 a). On y retrouve montés en parallèle un transistor MOS-P et un transistor MOS-N, ce qui permet de changer le sens de la polarité de la tension d'entrée. L'entrée COMMANDE et sa valeur complémentée (après l'inverseur) servent à OUVRIR ou à FERMER l'interrupteur. Quand le signal COMMANDE est au niveau HAUT, les deux transistors MOS CONDUISENT et l'interrupteur est fermé. Quand le signal COMMANDE est au niveau BAS, les deux transistors MOS sont BLOQUÉS et l'interrupteur est ouvert. Théoriquement, ce circuit fonctionne comme un relais électromécanique. En pratique, toutefois, lorsque l'interrupteur est fermé, le court-circuit n'est pas parfait; la résistance de l'interrupteur $R_{FERMÉ}$ est habituellement 200 Ω. La résistance de l'interrupteur lorsqu'il est ouvert est habituellement 10^{12} Ω, ce qui correspond à un circuit ouvert dans la plupart des cas. On utilise le symbole de la figure 8-45 b) pour représenter un interrupteur bilatéral.

On qualifie ce circuit de *bilatéral* parce que les bornes d'entrée et de sortie peuvent être interchangées. Les signaux appliqués à l'interrupteur peuvent être aussi bien numériques qu'analogiques, à condition qu'ils demeurent dans la gamme limitée par 0 et V_{DD} volts.

La figure 8-46 a) nous fait voir le schéma logique d'une puce contenant quatre interrupteurs bilatéraux 4016, également disponible dans la série 74HC sous le numéro 74HC4016. Les quatres interrupteurs bilatéraux de ce CI fonctionnent comme on vient de le décrire. Chaque interrupteur est contrôlé de façon indépendante par sa propre entrée de commande. Ainsi l'état MARCHE/ARRÊT de l'interrupteur du haut est commandé par l'entrée COMMANDE_A. Puisque ces interrupteurs sont bilatéraux, l'une ou l'autre des bornes sert d'entrée ou de sortie, comme nous en informe le libellé utilisé.

FIGURE 8-45 Interrupteur bilatéral CMOS (porte de transmission).



Commande	Interrupteur
0 V	Ouvert (OFF)
$+V_{DD}$	Fermé (ON)

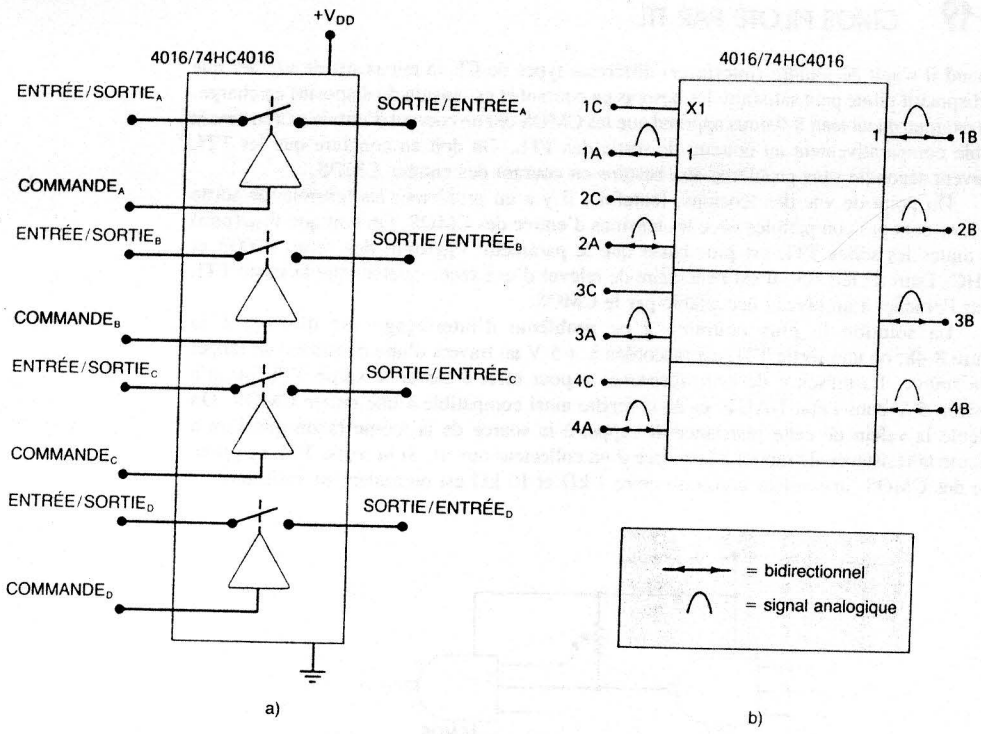


FIGURE 8-46 Quatre interrupteurs bilatéraux 4016/74HC4016: a) le symbole logique traditionnel; b) le symbole IEEE/ANSI.

Le symbole IEEE/ANSI de ce CI est reproduit à la figure 8-46 b). Chaque petit rectangle représente un interrupteur bilatéral, mais, comme toujours, seul le rectangle du haut contient la désignation de la notation fonctionnelle IEEE. Examinez ce rectangle, l'entrée de commande porte le libellé X1. Le « X » est le symbole signalant une entrée de commande qui dicte le sens de transmission entre les bornes entrée/sortie de même numéro. Le « 1 » du libellé X1 indique que cette broche de commande impose le sens de la transmission entre les bornes caractérisées par un libellé intérieur 1. À l'extérieur du rectangle, une ligne à double pointe dans une borne indique une opération bidirectionnelle. Le symbole en « fer à cheval » marque le fait que ces bornes acceptent des signaux analogiques en plus de signaux numériques.

EXEMPLE 8-18

Décrivez le fonctionnement du circuit de la figure 8-47.

Solution Il s'agit ici de deux interrupteurs bilatéraux connectés de manière à ce qu'un signal d'entrée commun analogique puisse être basculé vers une sortie X ou une sortie Y, selon l'état logique de l'entrée SÉLECTION SORTIE. Quand la valeur sur SÉLECTION SORTIE est BASSE, l'interrupteur du haut est fermé et celui du bas est ouvert, ce qui transmet $V_{entrée}$ à la sortie X. Quand la valeur sur la sortie de sélection est HAUTE,

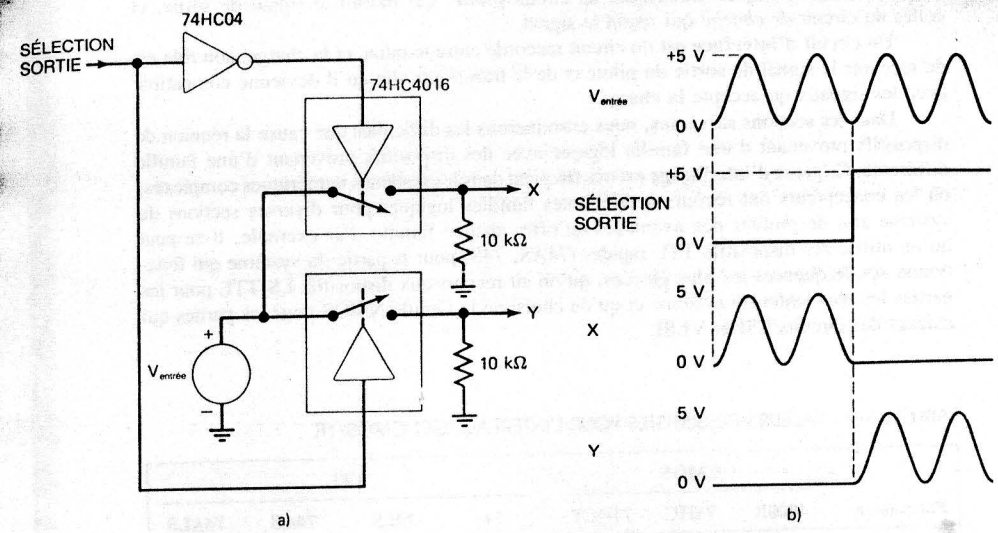


FIGURE 8-47 Exemple 8-18; des interrupteurs bilatéraux 74HC4016 utilisés pour basculer un signal analogique vers deux sorties différentes.

l'interrupteur du haut est ouvert et celui du bas est fermé, et alors le signal $V_{entrée}$ est transmis à la sortie Y. La figure 8-47 b) illustre certaines formes d'ondes types. Remarquez que ce circuit ne fonctionne comme décrit que si $V_{entrée}$ a une valeur comprise entre 0 et V_{DD}.

L'interrupteur bilatéral 4016/74HC4016 ne peut faire basculer que des tensions d'entrée comprises entre 0 et V_{DD}; il est donc impossible de s'en servir pour basculer des signaux à la fois positifs et négatifs par rapport à la masse. Les CI 4316 et 74HC4316 sont quatre interrupteurs bilatéraux qui peuvent commuter des signaux analogiques *bipolaires*. Ces dispositifs possèdent une seconde borne d'alimentation, appelée V_{EE}, qui peut être portée à un potentiel négatif par rapport à la masse. Cette borne permet donc aux signaux d'entrée de varier entre V_{EE} et V_{DD}. Par exemple, pour $V_{EE} = -5$ V et $V_{DD} = +5$ V, le signal analogique d'entrée pourra avoir n'importe quelle tension entre -5 V et $+5$ V.

QUESTIONS DE RÉVISION

1. Décrivez le fonctionnement d'un interrupteur bilatéral CMOS.
2. *Vrai ou faux:* Il n'existe pas d'interrupteur bilatéral en technologique TTL.

8-18 INTERFAÇAGE DES CI

Le terme *interfaçage* signifie la connexion des sorties d'un circuit ou d'un système aux entrées d'un autre circuit ou système dont les caractéristiques électriques sont différentes. Souvent, il est impossible de recourir à une connexion directe en raison de la différence

entre les caractéristiques électriques du circuit *pilote*, qui fournit le signal de sortie, et celles du circuit *de charge* qui reçoit le signal.

Un circuit **d'interface** est un circuit raccordé entre le pilote et la charge; son rôle est de recevoir le signal de sortie du pilote et de le transposer afin qu'il devienne compatible avec les signaux qu'accepte la charge.

Dans les sections suivantes, nous examinerons les difficultés que cause la réunion de dispositifs provenant d'une famille logique avec des dispositifs provenant d'une famille différente. Ce genre d'interfaçage est très fréquent dans les systèmes numériques complexes, où les concepteurs ont recours à différentes familles logiques pour diverses sections du système afin de profiter des avantages qu'offre chaque famille. Par exemple, il se peut qu'on utilise des dispositifs TTL rapides (74AS, 74S) pour la partie du système qui fonctionne aux fréquences les plus élevées, qu'on ait recours aux dispositifs LS-TTL pour les parties les plus lentes du système et qu'on choisisse la famille NMOS pour les parties qui exigent des circuits LSI et VLSI.

TABLEAU 8-9 VALEURS PESSIMISTES POUR L'INTERFAÇAGE* CMOS/TTL

	CMOS			TTL			
Paramètre	**4000B**	**74HC**	**74HCT**	**74**	**74LS**	**74AS**	**74ALS**
V_{IH}(min)	3,5 V	3,5 V	2,0 V	2,0 V	2,0 V	2,0 V	2,0 V
V_{IL}(max)	1,5 V	1,0 V	0,8 V	0,8 V	0,8 V	0,8 V	0,8 V
V_{OH}(min)	4,95 V†	4,9 V†	4,9 V†	2,4 V	2,7 V	2,7 V	2,7 V
V_{OL}(max)	0,05 V†	0,1 V†	0,1 V†	0,4 V	0,5 V	0,5 V	0,4 V
I_{IH}(max)	1 μA	1 μA	1 μA	40 μA	20 μA	200 μA	20 μA
I_{IL}(max)	1 μA	1 μA	1 μA	1,6 mA	0,4 mA	2 mA	100 μA
I_{OH}(max)	0,4 mA	4 mA	4 mA	0,4 mA	0,4 mA	2 mA	400 μA
I_{OL}(max)	0,4 mA	4 mA	4 mA	16 mA	8 mA	20 mA	8 mA

* Tension d'alimentation = 5 V.
† Les CMOS pilotant seulement des entrées CMOS.

Les CI d'une même série logique sont conçus pour être raccordés ensemble sans autre forme de considération, à la condition que la sortance pour chaque sortie ne soit pas dépassée. Quand vous connectez la sortie d'un CI à l'entrée d'un CI d'une famille différente ou d'une série différente de la même famille, vous devez, généralement, prendre en considération les tensions et les courants des deux éléments. On veut dire par là que vous devrez consulter les fiches techniques pour connaître les paramètres de tension et de courant de cette sortie et de cette entrée. Dans les sections suivantes, nous consulterons le tableau 8-9 au moment de notre étude de divers cas d'interfaçage. Ce tableau renferme les valeurs des paramètres d'entrée et de sortie des dispositifs standard des séries TTL et CMOS, pour le cas pessimiste. Il faut bien être conscient que ces valeurs s'appliquent à des dispositifs standard ne comprenant aucun circuit d'entrée ou de sortie spécial. Il se peut que ces valeurs soient bien différentes pour certains dispositifs, comme pour les tampons associés à des circuits spéciaux en vue d'accroître leurs capacités de tension et de courant. En outre, les courants d'entrées pourront différer de ces valeurs dans le cas de dispositifs dont une entrée externe est raccordée, par l'intérieur, à plus d'une porte. Bien que les valeurs qu'on trouve dans ce tableau conviennent à la grande majorité des dispositifs, il est toujours plus prudent de se référer aux fiches particulières des CI.

8-19 CMOS PILOTÉ PAR TTL

Quand il s'agit de joindre (interfacer) différents types de CI, le mieux est de vérifier que le dispositif pilote peut satisfaire les besoins en courant et en tension du dispositif en charge. Un examen du tableau 8-9 nous apprend que les CMOS ont un courant d'entrée extrêmement faible comparativement au courant de sortie des TTL. On doit en conclure que les TTL peuvent répondre sans problème aux besoins en courant des entrées CMOS.

Du point de vue des tensions, toutefois, il y a un problème: les tensions de sortie TTL ne sont pas compatibles avec les tensions d'entrée des CMOS. On voit que V_{OH}(min) de toutes les séries TTL est plus basse que le paramètre V_{IH}(min) des séries 4000B et 74HC. Dans de tels cas, il est nécessaire de relever d'une façon quelconque la sortie TTL pour l'amener à un niveau acceptable par le CMOS.

La solution la plus courante à ce problème d'interfaçage est illustrée à la figure 8-48, où une sortie TTL est raccordée à +5 V au travers d'une résistance de rappel à la source. La présence de cette résistance a pour effet d'élever la sortie TTL jusqu'à environ 5 V dans l'état HAUT, et de la rendre ainsi compatible à une entrée CMOS. On calcule la valeur de cette résistance de rappel à la source de la même façon que l'on a calculé la résistance de rappel à la source d'un collecteur ouvert. Si la sortie TTL ne pilote que des CMOS, une valeur comprise entre 1 kΩ et 10 kΩ est normalement suffisante.

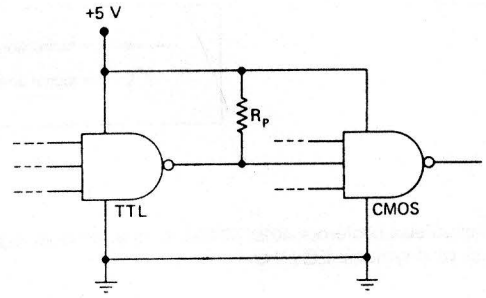


FIGURE 8-48 Utilisation d'une résistance de rappel à la source externe quand un TTL pilote un CMOS.

74HCT piloté par un TTL Comme nous l'avons déjà mentionné, la série 74HCT a été conçue pour être pilotée directement par des sorties TTL. Autrement dit, l'exigence pour la tension de sortie est la même que celle pour les dispositifs TTL standard. Voilà qui élimine la nécessité d'insérer la résistance de rappel à la source de la figure 8-48. La facilité de relier un TTL et un 74HCT se fait au détriment d'une consommation accrue dans l'élément CMOS.

CMOS haute tension piloté par un TTL Si le CI CMOS fonctionne sous une tension V_{DD} supérieure à 5 V, nous faisons face dans ce cas à une situation un peu plus difficile. Par exemple, si V_{DD} = 10 V, l'entrée CMOS doit avoir V_{IH}(min) = 7 V. Les sorties de la plupart des dispositifs TTL ne peuvent fonctionner à plus de 5 V, de sorte qu'il est tout à fait exclu de raccorder une résistance de rappel à +10 V. La série LS-TTL de certains fabricants (dont Fairchild) peut fonctionner avec une résistance de sortie branchée à 10 V. En général, il faut vérifier la fiche technique du dispositif avant de brancher une résistance sur une tension supérieure à 5 V.

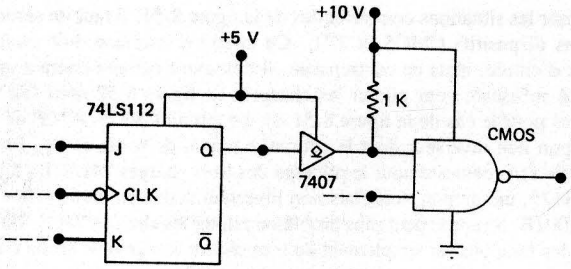


FIGURE 8-49 Utilisation d'un tampon à collecteur ouvert 7407 pour interfacer un TTL à un CMOS haute tension.

Quand il est impossible d'amener la sortie TTL jusqu'à V_{DD}, il faut se tourner vers d'autres solutions. L'une de celles-ci est illustrée à la figure 8-49, sur laquelle un tampon à collecteur ouvert 7407 agit comme interface entre une sortie en totem TTL et un CMOS fonctionnant à $V_{DD} > 5$ V. Le boîtier 7407 est la contrepartie non-inverseur du 7406 et la tension nominale de sa sortie est 30 V.

Une autre solution consiste à recourir à un circuit **transposeur de niveau** comme celui du boîtier 40104. Cette puce CMOS est conçue pour recevoir une entrée basse tension (comme la sortie d'un TTL) et pour la transformer en une sortie haute tension convenable pour un CMOS.

QUESTIONS DE RÉVISION

1. Dites ce qu'il faut faire pour interfacer une sortie TTL ordinaire à une entrée 4000B ou 74HC. Supposez $V_{DD} = +5$ V.
2. Donnez les diverses façons d'interfacer un TTL à un CMOS haute tension.

8-20 TTL PILOTÉ PAR CMOS

Avant de s'attaquer au problème de l'interfaçage entre des sorties CMOS et des entrées TTL, il peut être utile de revoir les caractéristiques de sortie CMOS pour les deux états logiques.

La figure 8-50 a) nous fait voir le circuit de sortie équivalent pour l'état HAUT. La R_{OUVERT} du transistor MOS-P raccorde la borne de sortie à V_{DD} (souvenez-vous que le transistor MOS-N est BLOQUÉ). Donc le circuit de sortie CMOS agit pratiquement comme une source V_{DD} ayant une résistance de source de R_{OUVERT}. La valeur caractéristique de R_{OUVERT} est entre 100 et 1000 ohms.

La figure 8-50 b) nous montre le circuit de sortie équivalent pour l'état BAS. La R_{OUVERT} du transistor MOS-N raccorde la borne de sortie à la masse (souvenez-vous que le transistor MOS-P est BLOQUÉ). Par conséquent, la sortie CMOS agit comme une faible résistance raccordée à la masse, c'est-à-dire comme un absorbeur de courant.

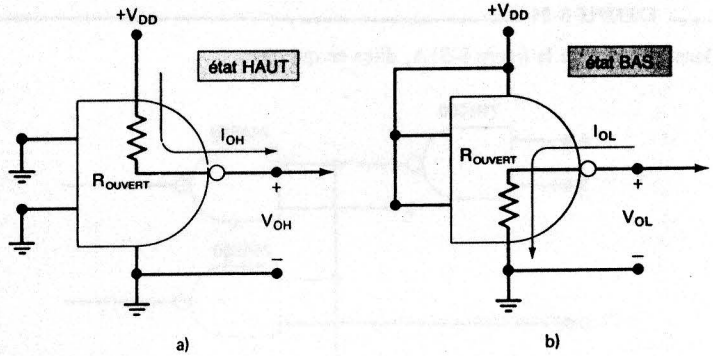


FIGURE 8-50 Circuits de sortie équivalents CMOS pour les deux états logiques.

TTL à l'état HAUT piloté par un CMOS Le tableau 8-9 nous apprend que les sorties CMOS fournissent suffisamment de tension (V_{OH}) pour répondre au besoin en courant d'entrée d'un TTL à l'état HAUT (V_{IH}). Il nous apprend aussi que les sorties CMOS délivrent beaucoup plus de courant qu'il n'en faut au TTL comme courant d'entrée (I_{IH}). Donc pour l'état HAUT, aucune considération spéciale ne doit être envisagée.

TTL à l'état BAS piloté par un CMOS Le tableau montre que les entrées TTL possèdent un courant d'entrée relativement intense dans l'état BAS, variant entre 100 μA et 2 mA. Les séries 74HC et 74HCT peuvent consommer jusqu'à 4 mA et ne manifestent alors aucune difficulté à piloter un *seul* TTL de n'importe quelle série. Par contre, la série 4000B est beaucoup moins versatile. Sa faible capacité de courant I_{OL} ne lui permet même pas de piloter une seule entrée des séries 74 ou 74AS.

EXEMPLE 8-19

Combien d'entrées 74LS une sortie 74HC parvient-elle à piloter? Même question pour une sortie 4000B.

Solution La série 74LS possède $I_{IL}(\text{max}) = 0{,}4$ mA. Le 74HC peut consommer jusqu'à $I_{OL}(\text{max}) = 4$ mA. Donc, une sortie 74HC peut piloter *dix* charges 74LS (4 mA/0,4 mA = 10).

Le 4000B ne peut consommer que 0,4 mA, de sorte qu'il ne peut piloter qu'*une* entrée 74LS.

EXEMPLE 8-20

Combien d'entrées 74ALS une sortie 74HC parvient-elle à piloter? Même question pour une sortie 4000B.

Solution Le courant $I_{IL}(\text{max})$ pour la série 74ALS est 100 μA. Donc, un 74HC peut piloter *quarante* entrées 74LS (4 mA/100 μA = 40). Une sortie 4000B peut piloter *quatre* entrées 74ALS (0,4 mA/100 μA).

EXEMPLE 8-21

Dans le circuit de la figure 8-51A, dites ce qui ne va pas.

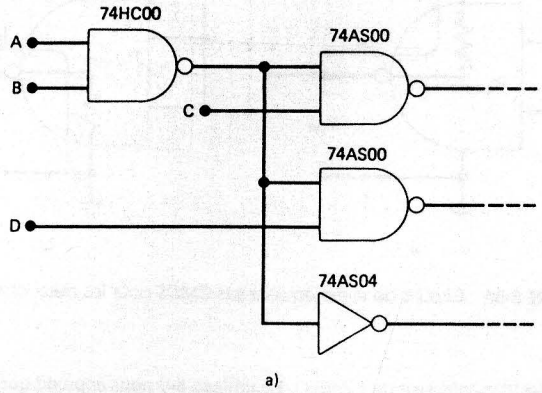


FIGURE 8-51A Exemple 8-21.

Solution Le 74HC00 peut consommer 4 mA, tandis que trois 74AS nécessitent l'absorption de 3 × 2 mA = 6 mA.

EXEMPLE 8-22

Dans le circuit de la figure 8-51B, dites ce qui ne va pas.

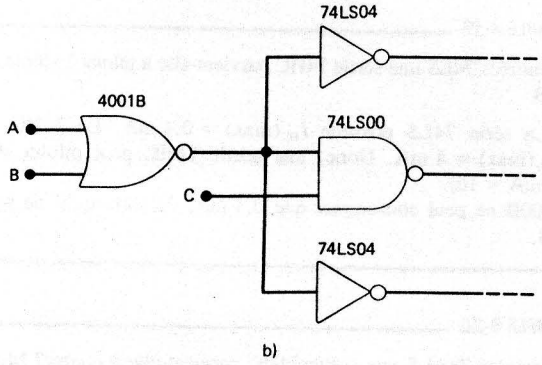


FIGURE 8-51B Exemple 8-22.

Solution La porte NON-ET 4001B peut absorber 0,4 mA, tandis que les trois entrées 74LS ont une exigence globale de 3 × 0,4 mA = 1,2 mA.

Pour les situations comme celles de la figure 8-51, il faut un certain circuit d'interface entre les dispositifs CMOS et TTL. Ce circuit d'interface doit avoir besoin d'un faible courant d'entrée, mais en contrepartie, il doit avoir comme courant nominal de sortie une intensité suffisante pour piloter les charges. La figure 8-52 nous fait voir deux interfaces possibles pour le cas de la figure 8-51 b). Le circuit CMOS 4050B de la figure 8-52 a) est un tampon non inverseur dont le courant nominal de sortie est $I_{OL}(\text{max}) = 3$ mA, ce qui le qualifie sans conteste pour le pilotage des trois charges 74LS. La figure 8-52 b) propose le 74LS125, un tampon trois états non inverseur activé en permanence, pouvant être piloté par le 4001B. Sa sortie peut sans problème piloter les charges 74LS. Dans ces deux circuits, le rôle des tampons est simplement de transmettre le signal de sortie du 4001B aux charges 74LS.

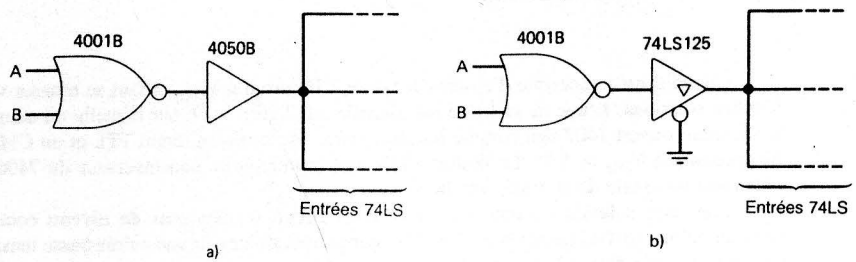
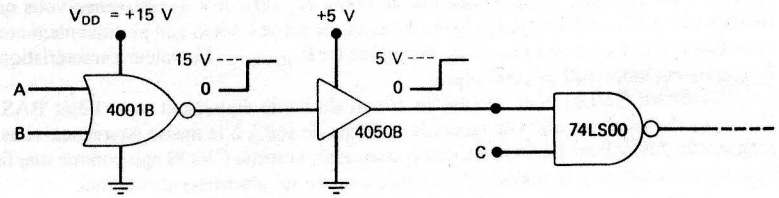


FIGURE 8-52 Tampons servant d'interface entre un CMOS faible courant et des entrées 74LS.

TTL piloté par un CMOS haute tension Certains fabricants de circuits intégrés ont produit plusieurs dispositifs TTL 74LS en mesure de supporter des tensions d'entrée pouvant aller jusqu'à 15 V. Ces dispositifs peuvent être pilotés directement par les sorties CMOS fonctionnant sous $V_{DD} = 15$ V. La plus grande partie des TTL ne peuvent soutenir une tension supérieure à 7 V sans se détériorer, d'où la nécessité d'utiliser une interface s'ils doivent être pilotés par un CMOS haute tension. Cette interface se comporte comme un **transposeur de niveau de tension** qui convertit la haute tension à l'entrée en une sortie de 5 V qui vient se connecter aux TTL. La figure 8-53 nous fait voir un 4050B qui agit comme transposeur de niveau de tension entre 15 V et 5 V.

FIGURE 8-53 Un tampon 4050B peut également servir comme transposeur de niveau de tension entre un CMOS et un TTL.



QUESTIONS DE RÉVISION

1. À quoi sert un circuit d'*interface*?
2. *Vrai ou faux:* Toutes les sorties CMOS peuvent piloter un TTL à l'état HAUT.
3. *Vrai ou faux:* Toute sortie CMOS peut piloter une seule entrée TTL.
4. Quelle série TTL peut piloter des sorties CMOS sans recourir à une résistance de rappel à la source?
5. Combien d'entrées 7400 une sortie 74HCT00 peut-elle piloter?
6. De quelle façon un CMOS haute tension est-il interfacé à un TTL?

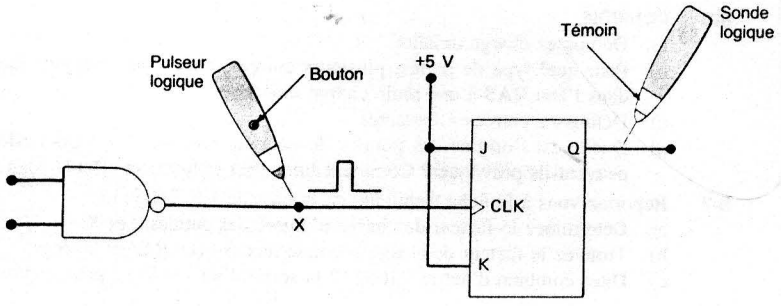
8-21 DÉPANNAGE

Un **pulseur logique** est un outil de dépannage et de contrôle qui génère des impulsions de courte durée quand on enfonce un bouton de commande. Le pulseur reproduit à la figure 8-54 possède une pointe en forme d'aiguille que l'on applique sur le noeud où l'on veut lancer une impulsion. Le pulseur logique a pour rôle de détecter le niveau de la tension actuelle et d'injecter une impulsion de tension de niveau opposé. Autrement dit, si le noeud est au niveau BAS, le pulseur envoie une étroite impulsion montante; par contre, si le noeud est au niveau HAUT, l'impulsion envoyée est descendante.

Cet outil sert à induire des changements momentanés du niveau logique à un noeud du circuit, et cela même si la sortie d'un autre circuit est connectée à ce noeud. À la figure 8-54, le pulseur est appliqué au noeud X, lui-même connecté à la sortie de la porte NON-ET. Le pulseur possède une impédance de sortie très faible (de façon caractéristique, 2 Ω ou moins); c'est pourquoi il dérive la sortie de la porte NON-ET et peut changer la tension à ce noeud. Cet instrument ne peut, toutefois, induire une impulsion de tension à un noeud court-circuité à la masse ou à V_{CC} (par ex. au travers un pont de soudure).

Test d'un circuit avec un pulseur et une sonde logique On peut utiliser un pulseur logique pour imposer manuellement une impulsion ou une série d'impulsions à un circuit en vue de tester sa réaction. Une sonde logique est presque toujours employée comme outil de surveillance de la réponse du circuit à l'action d'un pulseur logique. À la figure 8-54, le basculement de la bascule J-K est testé au moyen de l'application d'impulsions (par un pulseur) et d'une observation de ce qui se passe en Q avec une sonde logique. Le couple pulseur/sonde est d'une grande utilité pour vérifier le fonctionnement d'un élément logique câblé dans un circuit. Observez que le pulseur logique lance son impulsion dans le noeud du circuit sans déconnecter la sortie de la porte NON-ET qui pilote ce noeud.

FIGURE 8-54 Un pulseur logique peut injecter une impulsion dans n'importe quel noeud non court-circuité à la masse ou à V_{CC}.

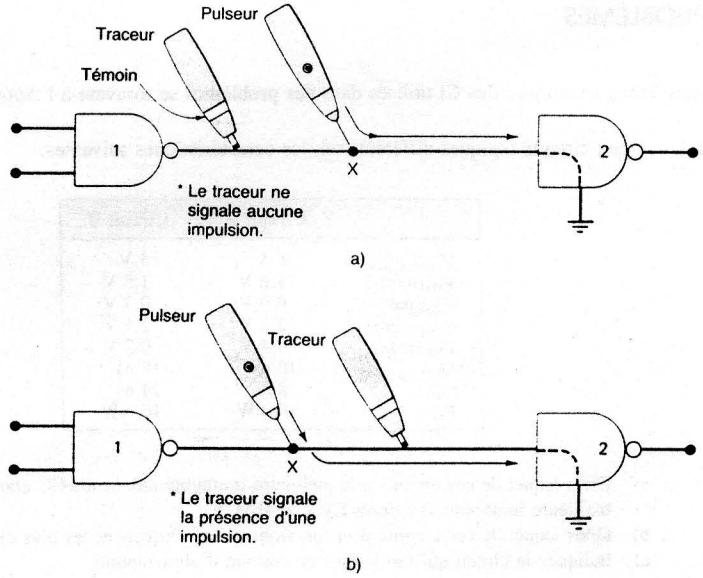


Dépistage des noeuds court-circuités On peut se servir du pulseur et de la sonde pour trouver les noeuds qui sont court-circuités à la masse ou à V_{CC}. Quand vous touchez avec la sonde et le pulseur le même noeud et que vous appuyez sur le bouton du pulseur, la sonde relève l'apparition d'une impulsion à ce noeud. Si la sonde mesure un niveau BAS constant, ce noeud est court-circuité à la masse; si elle mesure un niveau HAUT constant, ce noeud est court-circuité à V_{CC}.

Le traceur galvanométrique Un traceur galvanométrique est un instrument de dépannage qui peut déceler une *variation* de courant dans un fil ou un conducteur de plaquette de circuit imprimé sans endommager le circuit. Ce traceur est muni d'une pointe insolante renfermant une bobine de détection magnétique. Au moment où la pointe est posée sur le circuit, elle capte un champ magnétique variable généré par une variation de courant, ce qui provoque le clignotement d'une DEL. Le traceur galvanométrique est insensible aux niveaux de courant statiques, quelle qu'en soit l'intensité.

Un traceur est utilisé de concert avec un pulseur pour trouver l'emplacement exact des courts-circuits à la masse ou à V_{CC}. C'est d'ailleurs ce qu'illustre la figure 8-55, où le noeud X est court-circuité à la masse au travers d'un chemin direct interne dans l'entrée de la porte 2. Si le pulseur logique touche à X et que l'on appuie sur le bouton, aucune impulsion de tension n'est observée, à cause du court-circuit à la masse. Toutefois, il y

FIGURE 8-55 Utilisation combinée d'un traceur et d'un pulseur pour dépister les noeuds court-circuités.



aura une impulsion de courant de la sortie du pulseur jusqu'à la masse au travers du court-circuit. C'est cette impulsion de courant qu'est en mesure de déceler le traceur galvanométrique.

À la figure 8-55 a), le traceur est posé à la gauche de *X* et le pulseur à la droite. Le traceur n'indique aucune impulsion de courant dans le chemin qu'il surveille. À la figure 8-55 b), le traceur est placé à la droite de *X*. Cette fois-ci, au lancement de l'impulsion par le pulseur, le traceur détecte l'apparition d'une impulsion de courant à travers le chemin qu'il observe. Cela prouve que le court-circuit est interne et dans l'entrée de la porte 2 et non dans celle de la porte 1.

Le couple traceur galvanométrique/pulseur logique est très profitable pour le dépannage de circuits possédant de nombreux éléments à collecteur ouvert, à drain ouvert et à trois états connectés à un point commun et qui sont constamment au niveau HAUT ou BAS. N'importe laquelle des sorties peut être fautive, et la technique de la réunion d'un traceur et d'un pulseur permet d'isoler la sortie défectueuse.

QUESTIONS DE RÉVISION

1. Quelle est la fonction d'un pulseur logique?
2. *Vrai ou faux:* Un pulseur logique peut générer une impulsion de tension à n'importe quel noeud.
3. Quelle est la fonction d'un traceur galvanométrique?
4. *Vrai ou faux:* Un traceur galvanométrique ne détecte qu'un courant variable.

PROBLÈMES

(Les fiches techniques des CI utilisés dans ces problèmes se trouvent à l'Appendice II)

8-1 Deux circuits logiques différents ont les caractéristiques suivantes:

	Circuit A	Circuit B
$V_{\text{alimentation}}$	6 V	5 V
V_{IH}(min)	1,6 V	1,8 V
V_{IL}(max)	0,9 V	0,7 V
V_{OH}(min)	2,2 V	2,5 V
V_{OL}(max)	0,4 V	0,3 V
t_{PLH}	10 ns	18 ns
t_{PHL}	8 ns	14 ns
P_D	16 mW	10 mW

a) Dites lequel de ces circuits a la meilleure immunité aux bruits CC état BAS. La meilleure immunité aux bruits CC état HAUT.
b) Dites lequel de ces circuits peut fonctionner aux fréquences les plus élevées.
c) Indiquez le circuit qui tire le plus de courant d'alimentation.

8-2 Reportez-vous à l'Appendice II et dans les fiches techniques des CI, relevez les valeurs maximales nécessaires au calcul de P_D(moy), t_{pd}(max) et le produit vitesse-consommation pour une porte de chacun des CI TTL suivants.
a) 7432 b) 74S32 c) 74LS20 d) 74ALS20 e) 74AS20

8-3 Une certaine famille logique présente les paramètres de tension suivants:

V_{IH}(min) = 3,5 V V_{IL}(max) = 1,0 V
V_{OH}(min) = 4,9 V V_{OL}(max) = 0,1 V

a) Quelle pointe de tension positive maximale pourra-t-elle tolérer?
b) Quelle pointe de tension négative maximale pourra-t-elle tolérer?

CONTRÔLE

8-4 Pour chacun des énoncés, dites quelle caractéristique ou quel paramètre est décrit.
a) Le courant pénétrant dans une entrée portée au niveau 1.
b) Le courant tiré de la source V_{CC}(max) quand toutes les sorties sont BASSES.
c) Le temps nécessaire à une sortie pour basculer de 1 à 0.
d) Une mesure habituelle utilisée pour comparer la performance globale de différentes familles.
e) L'amplitude d'une pointe de tension tolérable sur une entrée HAUT et qui ne place pas l'élément en fonctionnement indéterminé.
f) Un boîtier de circuit intégré conçu pour être soudé directement sur la plaquette imprimée.
g) L'action d'une sortie BASSE qui absorbe du courant du dispositif qu'elle pilote.
h) Nombre d'entrées différentes qu'une sortie peut piloter en toute sécurité.
i) Montage de transistors de sortie dans un circuit TTL standard.
j) Expression désignant le transistor Q_4 d'un circuit de sortie TTL.
k) Fourchette des valeurs V_{CC} admissibles pour la série TTL 74.
l) V_{OH}(min) et V_{IH}(min) pour la série 74.
m) V_{IL}(max) et V_{OL}(max) pour la série 74.

SECTION 8-4

8-5 a) Dans le tableau 8-3, déterminez les marges de sensibilité aux bruits d'un dispositif 74LS pilotant une entrée 74ALS.
b) Même question qu'en a), mais pour un 74ALS pilotant un 74LS.
c) Quelle sera la marge de sensibilité aux bruits résultante d'un circuit logique qui combine des éléments 74LS et 74ALS?

SECTIONS 8-5 et 8-6

8-6 CONTRÔLE
a) Définissez charge unitaire.
b) Pour quel type de portes, plusieurs entrées reliées sont toujours équivalentes dans l'état BAS à une seule charge unitaire?
c) Définissez entrées « flottantes ».
d) Quelle est l'origine des pointes de courant dans un TTL? Quel effet néfaste peuvent-ils provoquer? Comment faire pour réduire cet effet?

8-7 Reportez-vous à la fiche technique de la bascule J-K 74LS112.
a) Déterminez le facteur de charge d'entrée des entrées *J* et *K*.
b) Trouvez le facteur de charge d'entrée aux entrées *RAZ* et horloge.
c) Dites combien d'autres 74LS112 la sortie d'un 74LS112 peut sychroniser.

8-8 La figure 8-56 a) nous montre une bascule J-K 74LS112 dont la sortie doit piloter 8 UL. Étant donné que cela dépasse la sortance du 74LS112, un certain tampon est nécessaire. La figure 8-56 b) illustre une solution: celle d'utiliser une des portes NON-ET du tampon 74LS37, comprenant quatre portes NON-ET, dont la sortance est plus élevée que celle du 74LS112. Notez que $\overline{Q}$ est utilisé puisque la porte NON-ET se comporte comme un INVERSEUR. Reportez-vous aux fiches techniques du 74LS37 et déterminez:

a) sa sortance;

b) le courant maximal absorbé dans l'état BAS.

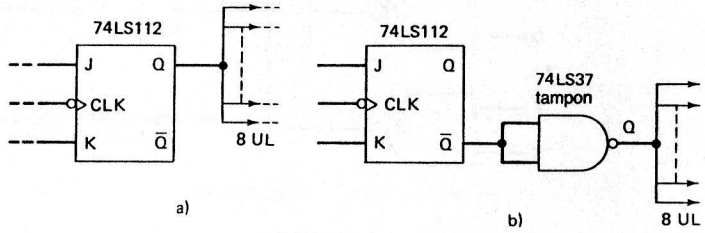
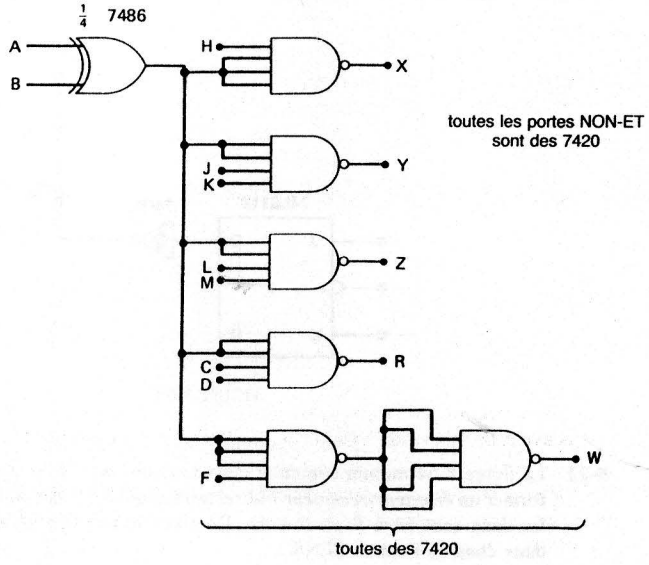


FIGURE 8-56

8-9 Les portes tampons sont généralement plus coûteuses que les portes ordinaires et parfois il y a plusieurs portes inutilisées qui permettraient de résoudre un problème de facteur de charge comme celui illustré à la figure 8-56 a). Montrez comment des portes NON-ET 74LS00 peuvent servir à résoudre ce problème.

8-10 Reportez-vous au schéma logique de la figure 8-57, dans lequel la sortie de la porte OU exclusif 7486 pilote plusieurs entrées 7420. Voyez si la sortance du 7486 est dépassée et expliquez.

FIGURE 8-57



8-11 Même problème que 8-10 mais avec les 74LS.

8-12 Dans le cas du circuit de la figure 8-57, calculez le temps maximal qui devrait s'écouler entre un changement appliqué à l'entrée *A* et sa répercussion à la sortie *W*. Utilisez les cas les plus pessimistes et les valeurs maximales des retards de propagation des portes. (*Suggestion*: Rappelez-vous que les portes NON-ET sont des portes inverseuses.)

8-13 a) La figure 8-58 nous montre un compteur 74193 dont l'entrée de réinitialisation maîtresse (vraie au niveau HAUT) est activée par un bouton-poussoir. La résistance *R* sert à garder au niveau BAS MR quand l'interrupteur est ouvert. Quelle est la valeur maximale pour *R*?

b) Même question, mais pour un 74LS193.

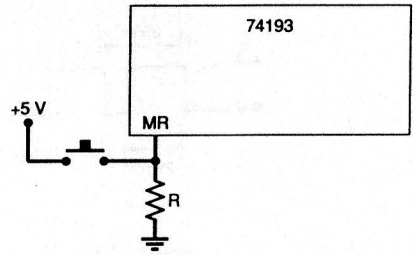


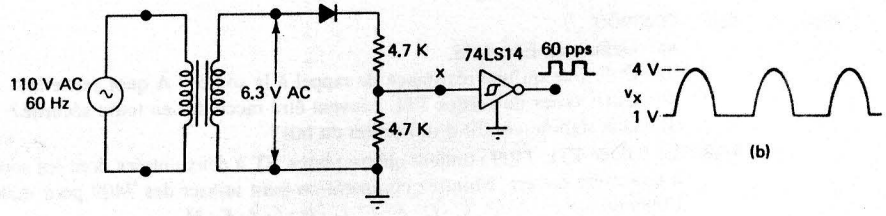
FIGURE 8-58

8-14 La figure 8-59 a) montre un circuit servant à convertir une onde sinusoïdale de 60 Hz en un signal de 60 ips qui peut déclencher correctement des bascules, des compteurs etc. C'est le genre de circuit qu'on peut retrouver dans une horloge numérique.

a) Décrivez le fonctionnement du circuit.

b) Un technicien teste ce circuit et observe que la sortie du 74LS14 demeure au niveau BAS. Il vérifie la forme d'onde de l'entrée de l'INVERSEUR et il observe l'onde de la figure 8-59 b). Croyant que l'INVERSEUR est défectueux, il le remplace sans que son observation change. Dites quel est d'après vous le problème et comment faire pour le régler. (*Suggestion:* Examinez attentivement la forme d'onde v_x.)

FIGURE 8-59



8-15 Pour chacune des formes d'ondes de la figure 8-60, trouvez *pourquoi* elle ne synchronise pas *correctement* une bascule 74LS112 (appliquée sur l'entrée *CLK*).

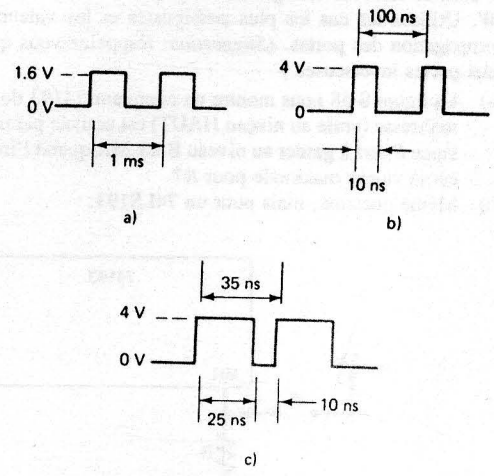


FIGURE 8-60

8-16 Une technicienne teste le montage expérimental d'un circuit logique. Pendant qu'elle teste le fonctionnement du circuit, elle s'aperçoit que plusieurs bascules et compteurs sont déclenchés de façon erratique. En bonne technicienne, elle vérifie la ligne V_{CC} avec un voltmètre continu et lit 4,97 V, ce qui est une valeur acceptable pour les TTL. Ensuite elle vérifie tout le câblage du circuit et remplace chaque CI un après l'autre, sans que cela corrige le problème. Finalement, elle décide d'utiliser son oscilloscope pour visionner l'onde V_{CC} et elle voit la forme de la figure 8-61.

Dites quelle est la cause probable du bruit affectant V_{CC}. Indiquez ce que la technicienne a oublié d'inclure quand elle a fait l'étude du montage expérimental du circuit.

FIGURE 8-61

SECTIONS 8-7 et 8-8

8-17 CONTRÔLE

a) Définissez ET câblé.
b) Qu'est-ce qu'une résistance de rappel à la source? À quoi sert-elle?
c) Quels types de sorties TTL peuvent être raccordés en toute sécurité?
d) Que signifie conflit d'utilisation du bus?

8-18 Le boîtier TTL 7409 contient quatre portes ET à deux entrées dont les sorties sont à collecteur ouvert. Montrez comment on peut utiliser des 7409 pour matérialiser l'opération $x = A \cdot B \cdot C \cdot D \cdot E \cdot F \cdot G \cdot H \cdot I \cdot J \cdot K \cdot M$.

8-19 Déterminez l'expression logique de la sortie X de la figure 8-62.

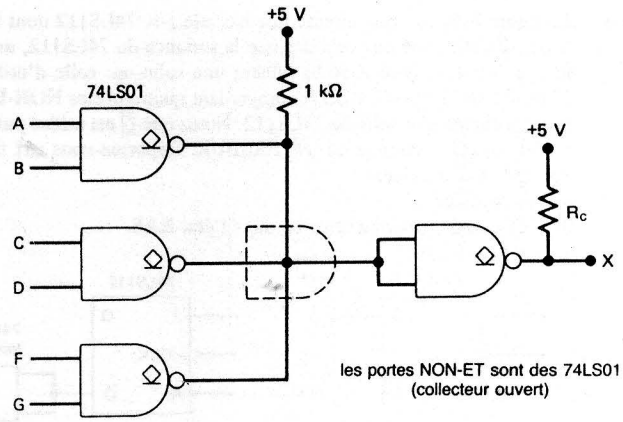


FIGURE 8-62

8-20 Calculez une valeur pour R_c de la figure 8-62 si la sortie X pilote les entrées *RAZ* de quatre bascules 74LS112.

8.21 La figure 8-63 nous fait voir un tampon inverseur à collecteur ouvert 7406 qui commande l'état ALLUMÉ/ÉTEINT d'une DEL indiquant l'état de la sortie Q d'une bascule. Les spécifications nominales de la DEL sont V_F = 2,4 V @ I_F = 20 mA, et I_F(max) = 30 mA.

a) Dites quelle est la tension à la sortie du 7406 quand $Q = 0$.
b) Choisissez pour la résistance en série une valeur convenant à un fonctionnement approprié.

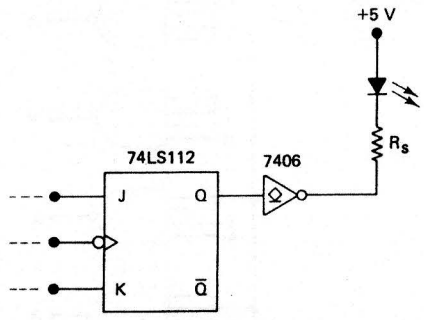


FIGURE 8-63

8-22 La figure 8-64 montre comment deux tampons trois états peuvent être réunis pour former un *émetteur-récepteur bidirectionnel* qui émet des données numériques dans les deux sens (A à B ou B à A). Décrivez le fonctionnement du circuit pour les deux états de l'entrée SENS.

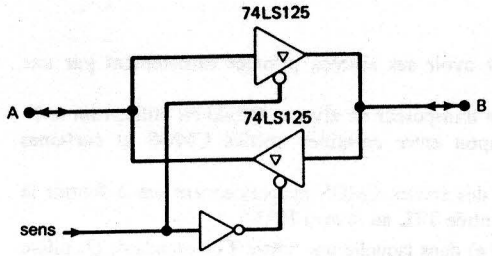


FIGURE 8-64

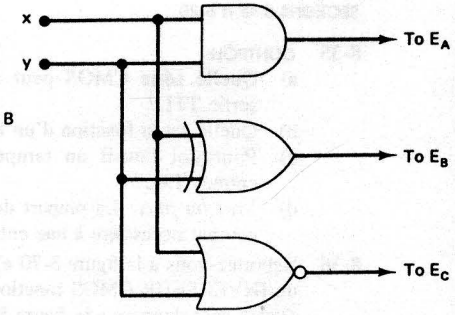


FIGURE 8-65

8-23 Le circuit de la figure 8-65 crée les entrées de validation appliquées aux circuits de la figure 8-29.

a) Déterminez laquelle des entrées de données (A, B ou C) sera transmise par le bus pour chacune des combinaisons des entrées x et y.

b) Indiquez pourquoi ce circuit ne fonctionnera plus si on remplace la porte NI par une porte NI EXCLUSIVE.

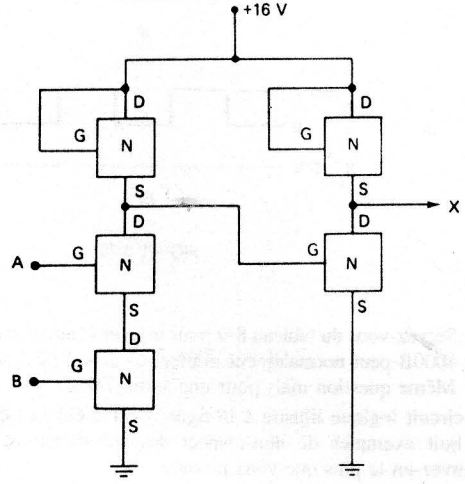
8-24 Une sortie 7406 pilote 5 UL de la famille TTL. Quelle est la résistance de rappel à la source la plus grande que l'on peut employer sans dégrader la marge de sensibilité aux bruits? (*Suggestion:* Voyez l'exemple 8-17 et la fiche technique du 7406.)

SECTIONS 8-10 à 8-113

8-25 Le circuit de la figure 8-66 est celui d'une porte logique NMOS. Déterminez le type de cette porte. Supposez que +16 V = 1 logique, 0 V = 0 logique.

8-26 Dessinez le circuit d'une porte OU NMOS.

FIGURE 8-66



8-27 Parmi les éléments suivants, lesquels sont des avantages de la logique PMOS et NMOS par rapport à la logique TTL?

a) Densité d'intégration plus grande
b) Vitesse de fonctionnement plus élevée
c) Sortance plus élevée
d) Plus appropriée aux circuits LSI
e) P_D plus faible
f) Sorties complémentaires
g) Immunité aux bruits supérieure
h) Procédé de fabrication plus simple
i) Utilisée dans un plus grand nombre de fonctions SSI et MSI
j) Requiert des tensions d'alimentation plus faibles

SECTIONS 8-14 et 8-15

8-28 Parmi les éléments suivants, lesquels sont des avantages des CMOS par rapport à la logique TTL?

a) Plus grande densité d'intégration
b) Plus rapide
c) Sortance plus élevée
d) Impédance de sortie plus faible
e) Processus de fabrication plus simple
f) Mieux adapté aux circuits LSI
g) P_D plus petit (sous 1 MHz)
h) Utilise seulement des transistors comme éléments de circuit
i) Capacité d'entrée plus petite

8-29 Déterminez laquelle des conditions de fonctionnement suivantes donnera probablement la plus petite dissipation de puissance moyenne (P_D) pour un système logique CMOS. Expliquez.

a) V_{DD} = 10 V, fréquence de commutation f_{max} = 1MHz
b) V_{DD} = 5 V, f_{max} = 10 kHz
c) V_{DD} = 10 V, f_{max} = 10kHz

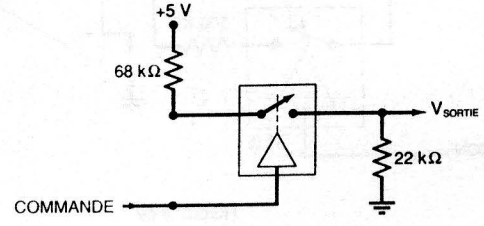
8-30 Quelles sont les marges de sensibilité aux bruits CC de la série 4000B fonctionnant sous une tension V_{DD} = 12 V.

8-31 Reportez-vous à la fiche technique de la porte NON-ET 74HC20 (Appendice II). Prenez les valeurs maximales et calculez P_D(moy), t_{pd}(moy) et le produit vitesse-consommation. Comparez avec les réponses trouvées au problème 8-2 pour un TTL.

SECTIONS 8-16 et 8-17

8-32 Déterminez les valeurs approximatives de V_{SORTIE} pour les deux états de l'entrée COMMANDE de la figure 8-67.

FIGURE 8-67



8-33 Déterminez la forme d'onde de la sortie X de la figure 8-68 pour les ondes d'entrées données. Supposez $R_{FERMÉ} \approx 200\ \Omega$ pour l'interrupteur bilatéral.

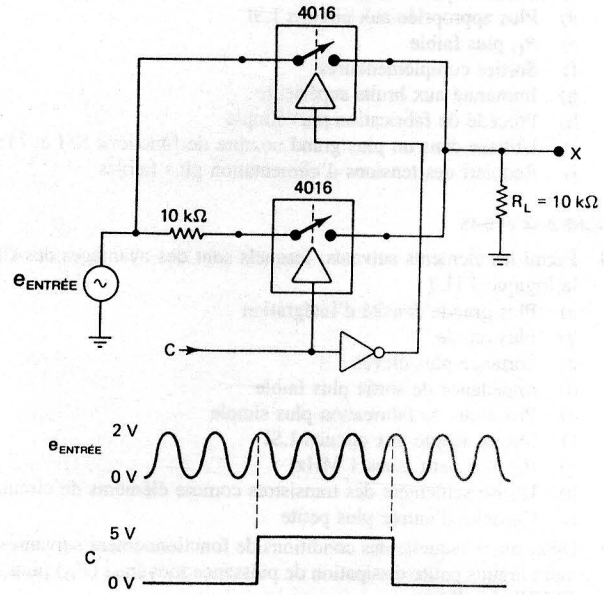


FIGURE 8-68

8-34 Trouvez le gain de l'amplificateur opérationnel de la figure 8-69 pour les deux états de l'entrée SÉLECTION GAIN. Ce circuit expose sommairement le principe de base de l'amplification de signaux commandés numériquement.

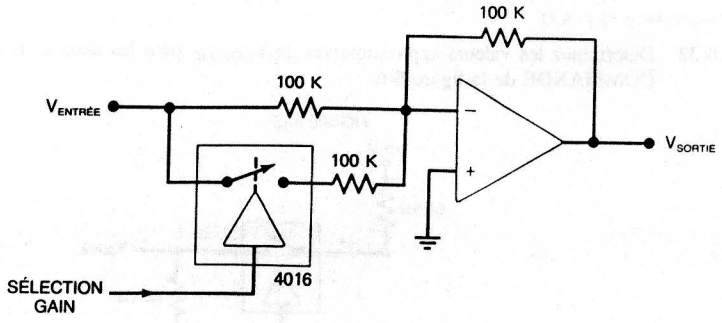


FIGURE 8-69

SECTIONS 8-18 à 8-20

8-35 CONTRÔLE

a) Quelle série CMOS peut avoir ses entrées pilotées directement par une sortie TTL?

b) Quelle est la fonction d'un transposeur de niveau. Quand en utilise-t-on un?

c) Pourquoi faut-il un tampon entre certaines sorties CMOS et certaines entrées TTL?

d) *Vrai ou faux:* La plupart des sorties CMOS ne parviennent pas à fournir le courant nécessaire à une entrée TTL au niveau HAUT.

8-36 Reportez-vous à la figure 8-70 a) dans laquelle une sortie TTL standard, Q, pilote un INVERSEUR CMOS fonctionnant sous V_{DD} = 10 V. Les formes d'ondes en Q et X sont données à la figure 8-70 b). Parmi les raisons suivantes qui expliquent pourquoi X demeure au niveau HAUT, choisissez celle qui est la plus plausible.

a) L'alimentation de 10 V est défectueuse.

b) La résistance de rappel à la source est trop grande.

c) La sortie 74LS112 passe bien en dessous de 10 V et maintient un niveau de 5,5 V dans l'état HAUT. Il s'agit de la gamme indéterminée de l'entrée CMOS.

d) L'entrée CMOS impose une charge qui abaisse la tension de sortie du TTL

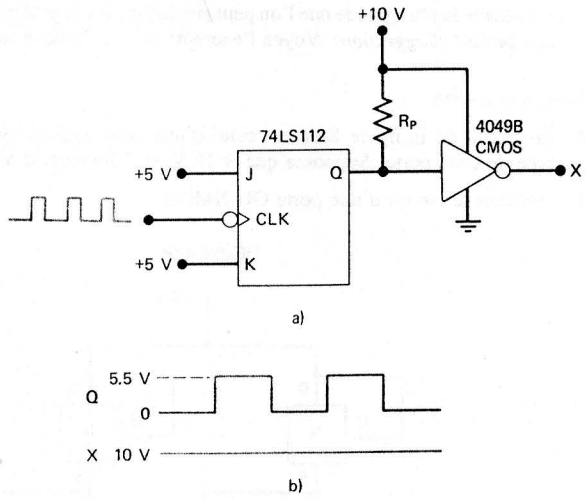


FIGURE 8-70

8-37 a) Servez-vous du tableau 8-9 pour trouver le nombre d'entrées 74AS qu'une sortie 4000B peut normalement piloter.

b) Même question mais pour une sortie 74HC.

8-38 Le circuit logique illustré à la figure 8-71 a été mal conçu. Il contient pas moins de huit exemples de non-respect des spécifications et caractéristiques des CI. Trouvez-en le plus que vous pourrez.

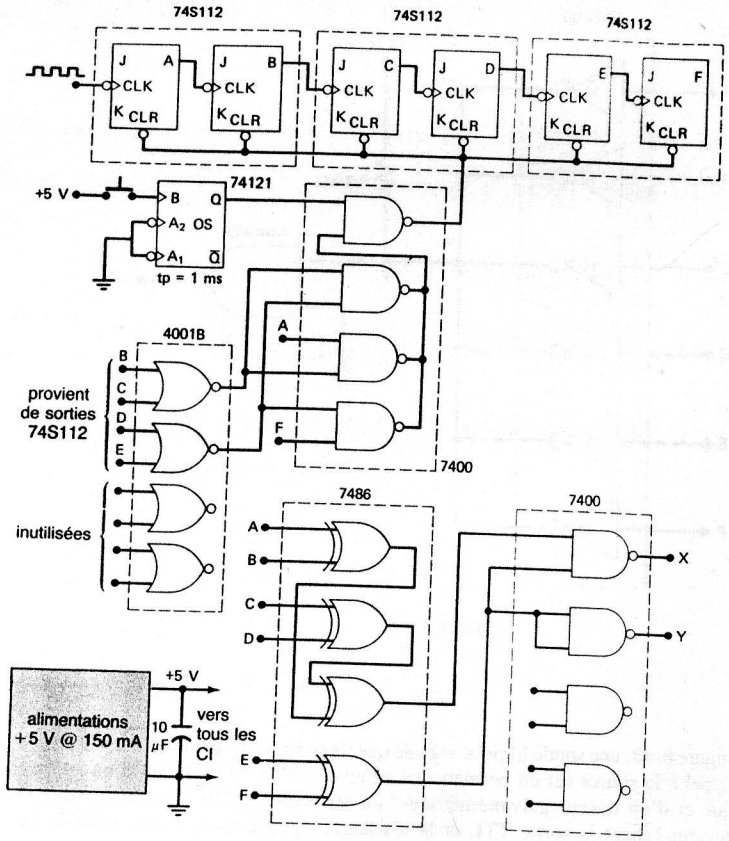


FIGURE 8-71

8-39 Même problème que 8-38, mais avec les nouvelles conditions que voici:
- chaque CI TTL est remplacé par son équivalent 74LS;
- le 4001B est remplacé par un 74HCT02.

SECTION 8-21

8-40 Le circuit de la figure 8-72 emploie un CI 74HC05 qui contient six INVERSEURS à drain ouvert. Ces inverseurs sont raccordés selon un montage ET câblé. Avec un pulseur et une sonde logiques, on constate que le noeud X est « gelé » sur le niveau HAUT. Décrivez la marche à suivre pour localiser cette défaillance avec un traceur.

8-41 Le circuit de la figure 8-54 possède un pont de soudure quelque part entre la sortie de la porte NON-ET et l'entrée de la bascule. Décrivez une procédure qui vous permettrait d'isoler l'emplacement de ce pont.

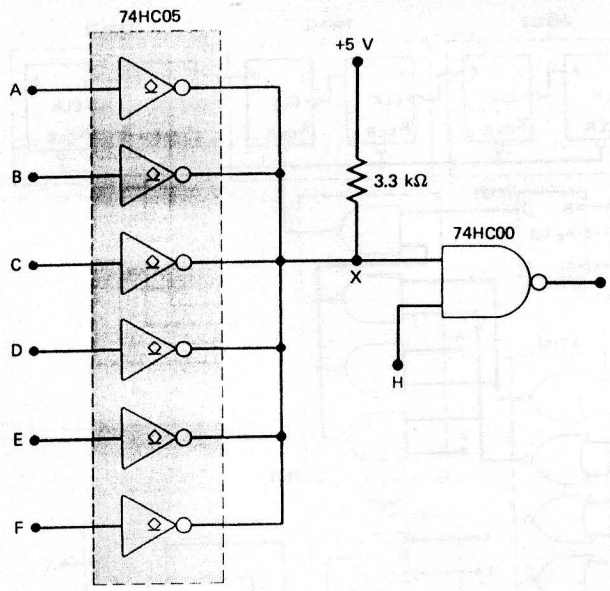


FIGURE 8-72

8-42 À la figure 8-48, une sonde logique signale que l'extrémité inférieure d'une résistance de rappel à la source est en permanence au niveau BAS. En s'aidant d'un pulseur logique et d'un traceur galvanométrique, un technicien relève un courant dans le conducteur reliant la sortie TTL et la résistance. Quelle est la cause possible de cette anomalie?
a) R_P est ouverte.
b) Le transistor à consommation de courant de la porte TTL a un court-circuit entre son émetteur et son collecteur.
c) Il y a rupture dans la connexion entre R_P et la porte CMOS.

8-43 À la figure 8-48, une sonde logique signale que l'extrémité inférieure de R_P est en permanence au niveau HAUT. On se sert du couple pulseur/traceur pour mettre en lumière qu'il y a un courant dans le conducteur entre R_P et la porte CMOS. Quelle est la cause possible de cette anomalie?

APPLICATION AU MICRO-PROCESSEUR

8-44 À la section 5-20, nous avons vu qu'un microprocesseur (MP), géré par un programme, transférait des données à des registres extérieurs. Le schéma de ce transfert est reproduit à la figure 8-73. Après le transfert des données dans le registre, ces dernières sont gardées dans cet élément en vue d'un usage ultérieur. Parfois, chaque bit individuel du registre a un rôle particulier. Par exemple, dans un calculateur d'automobile, chaque bit pourra représenter l'état de différentes grandeurs surveillées

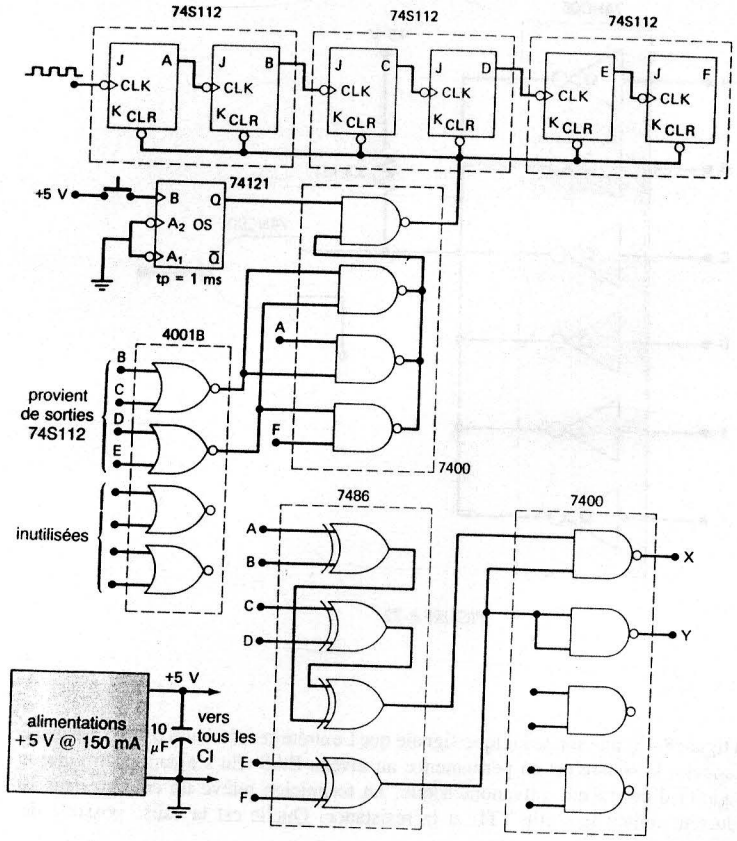


FIGURE 8-71

8-39 Même problème que 8-38, mais avec les nouvelles conditions que voici:
- chaque CI TTL est remplacé par son équivalent 74LS;
- le 4001B est remplacé par un 74HCT02.

SECTION 8-21

8-40 Le circuit de la figure 8-72 emploie un CI 74HC05 qui contient six INVERSEURS à drain ouvert. Ces inverseurs sont raccordés selon un montage ET câblé. Avec un pulseur et une sonde logiques, on constate que le noeud X est « gelé » sur le niveau HAUT. Décrivez la marche à suivre pour localiser cette défaillance avec un traceur.

8-41 Le circuit de la figure 8-54 possède un pont de soudure quelque part entre la sortie de la porte NON-ET et l'entrée de la bascule. Décrivez une procédure qui vous permettrait d'isoler l'emplacement de ce pont.

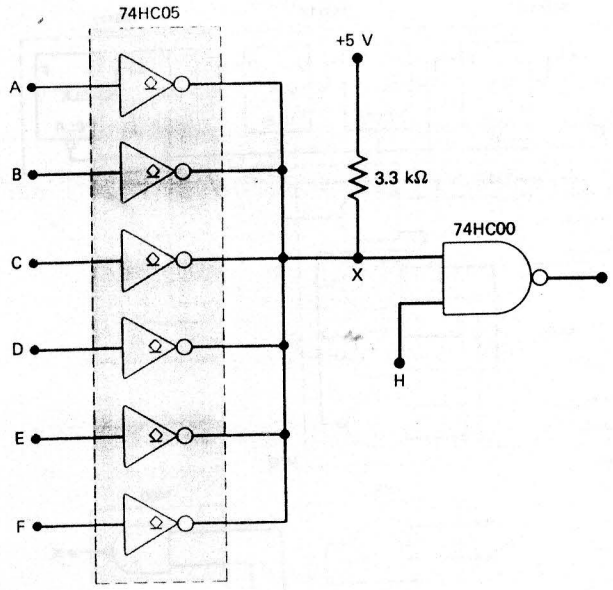


FIGURE 8-72

8-42 À la figure 8-48, une sonde logique signale que l'extrémité inférieure d'une résistance de rappel à la source est en permanence au niveau BAS. En s'aidant d'un pulseur logique et d'un traceur galvanométrique, un technicien relève un courant dans le conducteur reliant la sortie TTL et la résistance. Quelle est la cause possible de cette anomalie?
a) R_P est ouverte.
b) Le transistor à consommation de courant de la porte TTL a un court-circuit entre son émetteur et son collecteur.
c) Il y a rupture dans la connexion entre R_P et la porte CMOS.

8-43 À la figure 8-48, une sonde logique signale que l'extrémité inférieure de R_P est en permanence au niveau HAUT. On se sert du couple pulseur/traceur pour mettre en lumière qu'il y a un courant dans le conducteur entre R_P et la porte CMOS. Quelle est la cause possible de cette anomalie?

APPLICATION AU MICRO-PROCESSEUR

8-44 À la section 5-20, nous avons vu qu'un microprocesseur (MP), géré par un programme, transférait des données à des registres extérieurs. Le schéma de ce transfert est reproduit à la figure 8-73. Après le transfert des données dans le registre, ces dernières sont gardées dans cet élément en vue d'un usage ultérieur. Parfois, chaque bit individuel du registre a un rôle particulier. Par exemple, dans un calculateur d'automobile, chaque bit pourra représenter l'état de différentes grandeurs surveillées

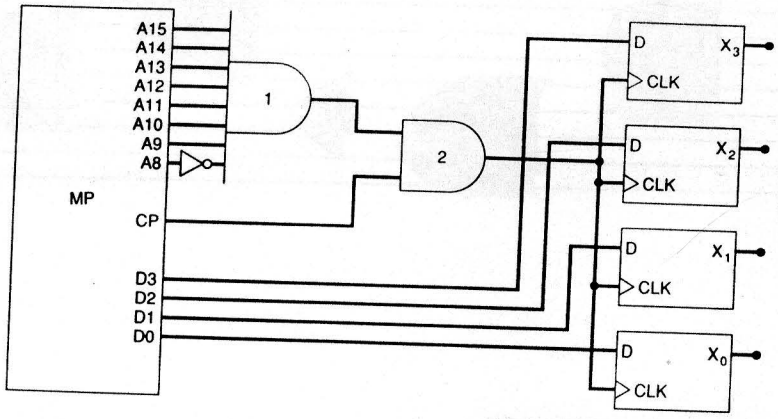


FIGURE 8-73

par le microprocesseur. Un bit pourra signaler une surchauffe du moteur, un autre, une pression d'huile trop basse. Dans d'autres applications, ces bits du registre serviront comme donnée pour obtenir un signal analogique nécessaire à la commande d'un dispositif analogique qui réagit à plusieurs niveaux de tension.

La figure 8-74 nous fait voir comment exploiter un microprocesseur pour générer une tension analogique à partir de la donnée contenu dans un registre, comme celui de la figure 8-73. Ensuite, ce signal analogique est délivré aux entrées d'un amplificateur de sommation. Supposons que le microprocesseur exécute un programme qui lui demande de transférer dans le registre à toutes les 10 μs les données regroupées dans le tableau qui suit. Tracez la forme d'onde résultante pour V_{SORTIE}.

Temps (μs)	Donnée du MP
0	0000
10	0010
20	0100
30	0111
40	1010
50	1110
60	1111
70	1111
80	1110
90	1100
100	1000

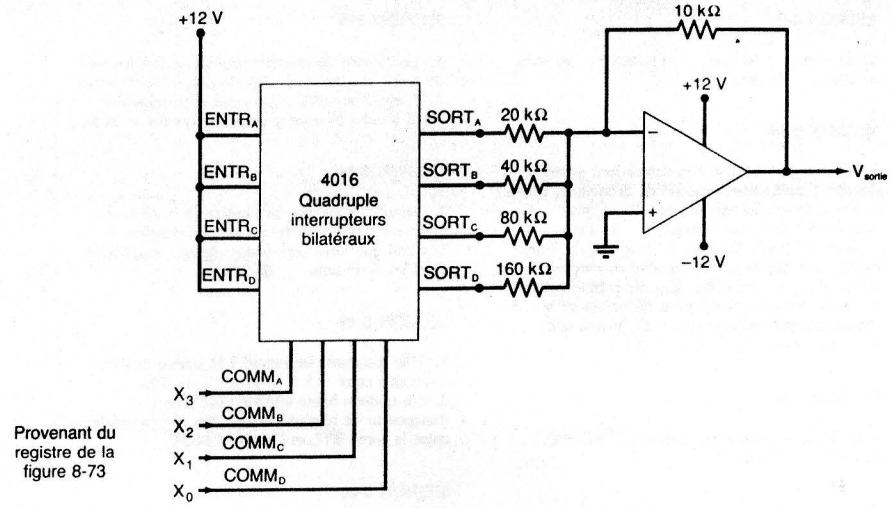


FIGURE 8-74 Problème 8-44.

RÉPONSE AUX QUESTIONS DE RÉVISION

SECTION 8-1

1. Voyez texte. **2.** Faux. **3.** V_{NH} est la différence entre V_{OH}(min) et V_{IH}(min); V_{NL} est la différence entre V_{IL}(max) et V_{OL}(max). **4.** Faux. **5.** Absorption de courant: une sortie absorbe (consomme) en réalité le courant de l'entrée du circuit qu'elle pilote. Injection de courant: une sortie injecte (délivre) le courant à l'entrée qu'elle pilote. **6.** Montage en surface. **7.** Boîtier plat.

SECTION 8-2

1. Vrai. **2.** BASSE. **3.** Temps de commutation rapide, faible consommation. Importante pointe de courant durant le basculement de BAS à HAUT. **4.** Q_3.

SECTION 8-4

1. a) 74AS **b)** 74S, 74LS; **c)** 74H, 74L; **d)** 74S, 74LS, 74AS, 74ALS; **e)** 74ALS. **2.** Tous les trois fonctionnent à 40 MHz, mais le 74ASL193 consomme moins d'énergie.

SECTION 8-5

1. Résistance de l'état conducteur de Q_4 et V_{OL}(max). **2.** 200 μA. **3.** Ses tensions de sortie ne peuvent rester dans les plages admissibles du 0 logique et du 1 logique. **4.** Deux.

SECTION 8-6

1. BAS. **2.** Connecter $+V_{CC}$ à travers une résistance de 1 kΩ; connecter l'entrée à une autre entrée utilisée. **3.** Connecter à la masse; connecter à une autre entrée utilisée. **4.** Faux; seulement dans l'état BAS. **5.** En connectant de petits condensateurs RF entre V_{CC} et la masse à proximité de chaque CI TTL afin de filtrer les pointes de tension dues aux variations rapides de courant durant le basculement de BAS à HAUT.

SECTION 8-7

1. En connectant les sorties TTL ensemble. **2.** Le courant destructeur peut circuler quand une sortie est HAUTE pendant que l'autre est BASSE. **3.** Le collecteur de Q_4, le transistor d'absorption, n'est pas branché (il n'y a pas de Q_3). **4.** Pour produire un niveau V_{OH}. **5.** La valeur nominale I_{OL} de la sortie et le nombre de charges pilotées. **6.** Aucun transistor Q_3 de rappel à la source actif. **7.** Voyez figure 8-26.

SECTION 8-8

1. HAUT, BAS, Haute impédance. **2.** Haute impédance. **3.** Quand au moins deux sorties trois états, connectées au même bus, sont activées simultanément. **4.** $E_A = E_B = 0, E_C = 1$. **5.** Voyez figure 8-30.

SECTION 8-9

1. a) Vrai; **b)** vrai; **c)** faux; **d)** vrai; **e)** faux; **f)** faux.

SECTION 8-13

1. Avantages — faible consommation; grande densité d'intégration; procédé de fabrication simple et peu coûteux. Inconvénients — lent, plus sensible à l'électricité statique. **2.** Les capacités d'entrée des portes; chaque ajout d'une charge peut augmenter la capacité de charge. **3.** Un R_{SORTIE} et un C_{CHARGE} de grande valeur. **4.** Haute densité d'intégration découlant de la structure simple du circuit. **5.** Voyez texte. **6.** Voyez texte.

SECTION 8-14

1. a) Faux — consomme moins; **b)** vrai; **c)** faux; **d)** vrai.

SECTION 8-15

1. 74C, 74HC, 74HCT. **2.** Vitesse supérieure, courant de sortie plus fort. **3.** Peut être piloté directement par une sortie TTL. **4. a)** Augmente; **b)** diminue; **c)** augmente. **5.** Les condensateurs des entrées CMOS doivent être chargés par la sortie de la source de signal. **6.** Connecté à la masse, V_{DD}, ou à une autre entrée utilisée. **7.** Voyez section 8-13, dernier paragraphe. **8.** Vérifiez le tableau 8-8, consommation par porte, et prenez note que la série 4000B a la plus faible consommation. **9.** Dans le tableau 8-8, la série 4000B manifeste la marge de sensibilité aux bruits la plus large.

SECTION 8-16

1. Le courant destructeur peut passer; la tension de sortie se retrouvera dans la plage indéterminée. **2.** L'étage de sortie est formée d'un transistor MOS à canal N dont le drain n'est pas connecté.

SECTION 8-17

1. Le niveau logique de l'entrée de commande impose l'état ouvert/fermé de l'interrupteur bilatéral qui laisse passer des signaux analogiques dans les deux sens. **2.** Vrai.

SECTION 8-19

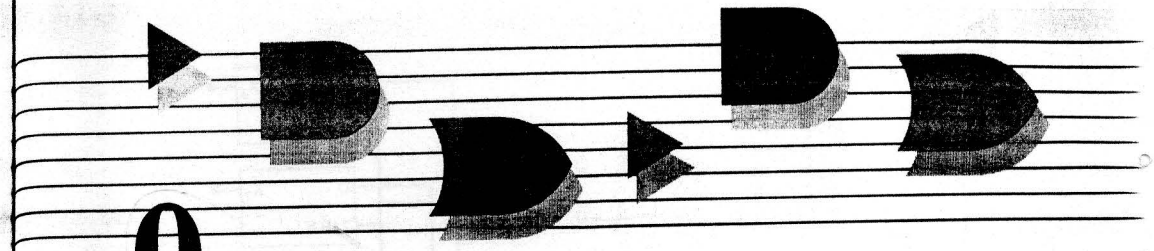
1. Une résistance de rappel à la source doit être intercalée entre +5 V et la sortie du TTL. **2.** Un tampon haute tension (7407) ou un transposeur de niveau (40104) doit être intercalé entre la sortie TTL et l'entrée CMOS.

SECTION 8-20

1. Il prélève la sortie d'un circuit pilote et le conforme afin de le rendre compatible avec les exigences d'entrée de la charge. **2.** Vrai. **3.** Faux — par exemple, la série 4000B ne peut absorber I_{IL} d'un 74 ou d'un élément 74AS. **4.** Aucun. **5.** Deux. **6.** Au travers d'un transposeur de niveau comme le 4050B.

SECTION 8-21

1. Utilisé pour injecter une impulsion de polarité choisie dans un noeud non court-circuité à V_{CC} ou à la masse. **2.** Faux. **3.** Indique la présence d'une impulsion de courant. **4.** Vrai.

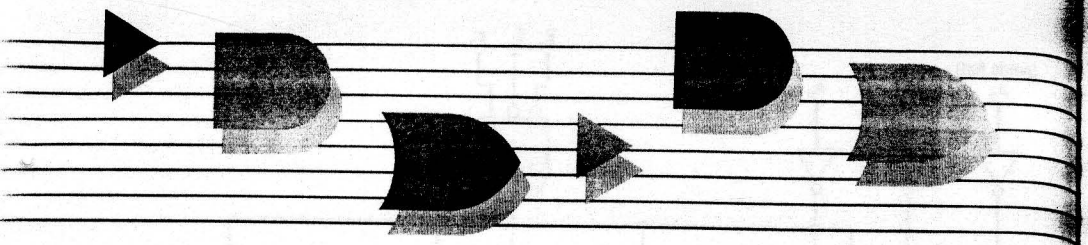


9

CIRCUITS LOGIQUES MSI

Contenu

9-1 Décodeurs
9-2 Pilotes/décodeurs DCB-7 segments
9-3 Afficheurs à cristaux liquides
9-4 Codeurs
9-5 Symboles IEEE/ANSI
9-6 Dépannage
9-7 Multiplexeurs (sélecteurs de données)
9-8 Applications des multiplexeurs
9-9 Démultiplexeurs (distributeurs de données)
9-10 Symbologie IEEE/ANSI: complément
9-11 Supplément de dépannage
9-12 Comparateur de grandeurs
9-13 Convertisseurs de code
9-14 Registres trois états
9-15 Bus de donnée



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Analyser et installer des décodeurs et des codeurs dans divers types d'applications numériques.
- Comparer les mérites respectifs (avantages et inconvénients) des afficheurs DEL et des ACL.
- Mettre en pratique la technique observation et raisonnement dans le dépannage de circuits numériques.
- Décrire le fonctionnement des multiplexeurs et des démultiplexeurs en analysant plusieurs applications de circuits.
- Comparer deux nombres binaires en recourant à un circuit comparateur de grandeurs.
- Indiquer les précautions à prendre quand on raccorde des circuits numériques à un bus commun.
- Traduire le sens de la notation utilisée dans les symboles IEEE/ANSI pour décrire différents éléments MSI.

Introduction

Dans les systèmes numériques, on retrouve toujours des données et des informations codées sous forme binaire qui sont sans cesse soumises à des opérations, comme: 1) le *décodage et le codage* — transposition des données d'un code à un autre, 2) *le multiplexage* — choix d'un groupe de données parmi plusieurs, 3) le *démultiplexage* — aiguillage des données vers une destination parmi plusieurs, et 4) l'*acheminement par bus* — transmission de données entre plusieurs dispositifs par l'intermédiaire d'un bus commun. Toutes ces opérations, ainsi que d'autres, sont désormais faciles à matérialiser grâce aux nombreux circuits intégrés de la classe MSI (intégration à moyenne échelle).

Dans le présent chapitre, nous nous proposons d'étudier plusieurs types de dispositifs MSI parmi les plus courants. Pour chaque type de dispositif, nous commencerons par une courte discussion de son principe de fonctionnement, puis nous présenterons des CI spécifiques. Ensuite, nous montrerons quel rôle ces dispositifs jouent dans diverses applications quand on les utilise soit seuls, soit combinés à d'autres circuits intégrés.

9-1 DÉCODEURS

Le *décodeur* est un circuit logique qui établit la correspondance entre un code d'entrée binaire de N-bits et M* lignes de sortie; pour chacune des combinaisons possibles des entrées, une seule ligne de sortie est validée. La figure 9-1 nous fait voir le schéma général d'un décodeur ayant N entrées et M sorties. Étant donné que chacune des N entrées peut être soit 0 soit 1, il y a 2^N combinaisons ou codes d'entrée possibles. Pour chacune des combinaisons d'entrée possibles, une seule des M sorties passera au niveau HAUT; toutes les autres sorties demeureront au niveau BAS. De nombreux décodeurs sont conçus pour avoir des sorties vraies au niveau BAS, c'est-à-dire que seule la sortie choisie est au niveau BAS tandis que les autres demeurent au niveau HAUT. Quand cette convention s'applique, il y a toujours sur les lignes de sortie du schéma des petits ronds.

Certains décodeurs n'utilisent pas toute la gamme des 2^N codes d'entrée possibles, mais seulement un sous-ensemble de celle-ci. Par exemple, un décodeur DCB-décimal a comme entrée un code binaire de 4 bits et *dix* lignes de sortie, une pour chacune des *dix* représentations du code DCB, 0000 à 1001. Souvent, les décodeurs de ce genre sont conçus de façon à ce que les codes inutilisés n'activent *aucune* des sorties lorsqu'ils se retrouvent appliqués sur l'entrée.

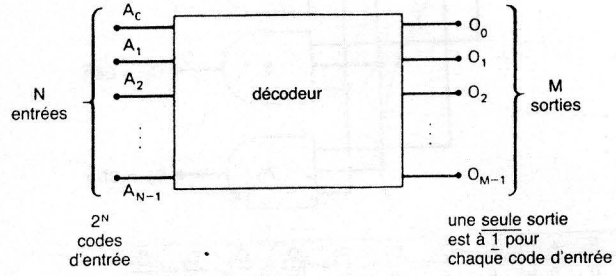
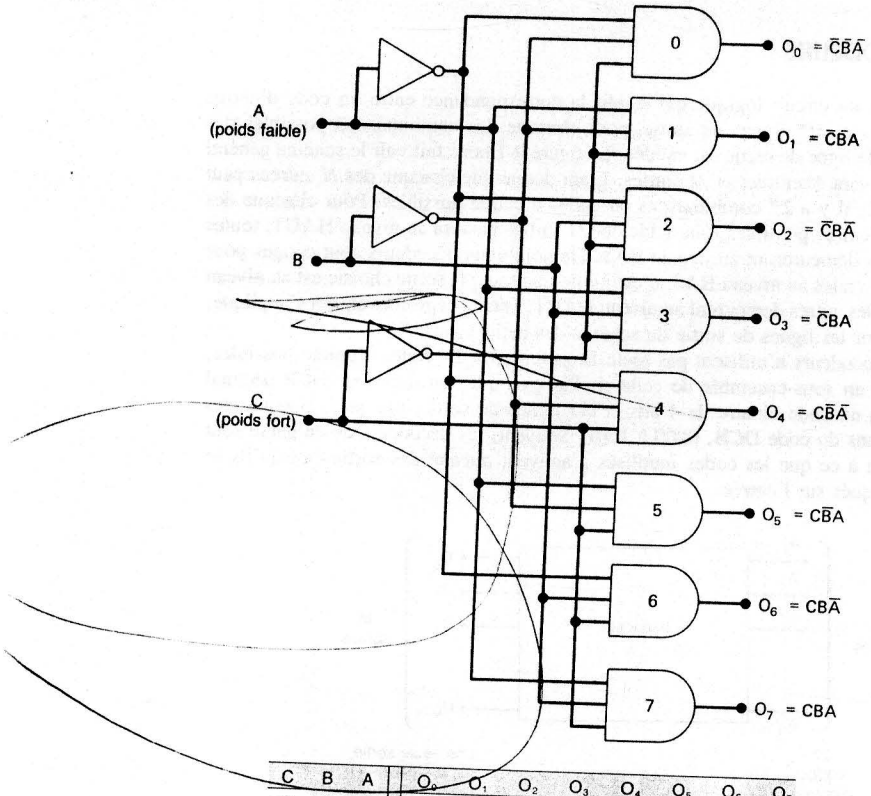


FIGURE 9-1 Schéma général d'un décodeur.

Au chapitre 7, nous avons vu comment on parvenait à détecter les divers états du compteur en lui associant un décodeur. Dans cette application, le code binaire appliqué aux entrées du décodeur provenait des sorties des bascules du compteur. Les mêmes circuits de décodage de base sont utilisés d'où que proviennent les entrées. La figure 9-2 nous fait voir les circuits d'un décodeur ayant trois voies d'entrée et $2^3 = 8$ voies de sortie. Sur ce schéma il n'y a que des portes ET, de sorte que les sorties sont vraies au niveau HAUT. Pour avoir des sorties vraies au niveau BAS, il faut utiliser des portes NON-ET. Remarquez que pour un code d'entrée donné, la seule sortie qui est vraie (HAUT) est celle correspondant à l'équivalent décimal du code d'entrée binaire (par exemple la sortie S_6 passe à 1 quand $CBA = 110_2 = 6_{10}$).

Il y a plusieurs façons de désigner ce décodeur. On peut l'appeler *décodeur entrée trois voies, sortie huit voies*, parce qu'il y a trois voies d'entrée et huit voies de sortie. On peut aussi dire que c'est un *décodeur ou convertisseur binaire-octal* parce qu'il établit la correspondance entre un code d'entrée binaire de 3 bits et une sortie parmi huit (octal). On dit également que c'est un *décodeur 1 parmi 8*, parce qu'il y a 1 sortie parmi les 8 qui est activée à la fois.

* N est un entier quelconque et M est un entier inférieur ou égal à 2^N.

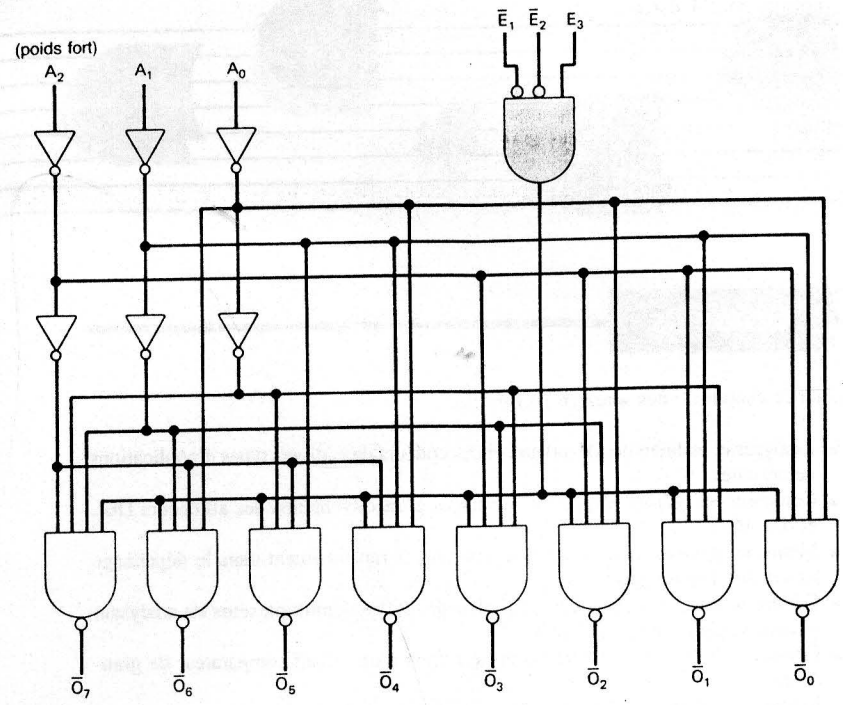


C	B	A	O_0	O_1	O_2	O_3	O_4	O_5	O_6	O_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

FIGURE 9-2 Décodeur entrée trois voies, sortie huit voies (ou 1 parmi 8).

Entrées VALIDATION Certains décodeurs sont dotés d'une ou de plusieurs entrées VALIDATION qui servent à commander son fonctionnement. Par exemple, reportez-vous au codeur illustré à la figure 9-2 et imaginez qu'il y ait une ligne commune VALIDATION raccordée à une quatrième entrée de chaque porte. Quand cette ligne est gardée au niveau HAUT, le décodeur fonctionne normalement et le code d'entrée *A*, *B*, *C* détermine quelle sortie passe au niveau HAUT. Quand VALIDATION est gardée au niveau BAS, *toutes* les sorties sont forcées dans l'état BAS quels que soient les niveaux appliqués aux entrées *A*, *B*, *C*. Donc ce décodeur est VALIDÉ seulement si le signal VALIDATION est au niveau HAUT.

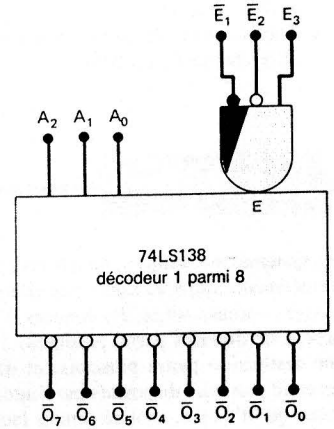
La figure 9-3 a) nous montre le schéma logique du décodeur 74LS138 tel que le donne le manuel des fiches techniques TTL de Fairchild. L'examen attentif de ce schéma



a)

$\overline{E}_1$	$\overline{E}_2$	E_3	sorties
0	0	1	réagit au code d'entrée $A_2A_1A_0$
1	X	X	invalidées – toutes au niveau HAUT
X	1	X	invalidées – toutes au niveau HAUT
X	X	0	invalidées – toutes au niveau HAUT

b)



c)

FIGURE 9-3 a) Schéma logique d'un décodeur 74LS138; b) sa table de vérité; c) son symbole logique. (Gracieuseté de Fairchild, filiale Schlumberger).

nous permet de déterminer exactement le fonctionnement de ce décodeur. D'abord, remarquez que ses sorties sont celles de portes NON-ET, par conséquent elles sont vraies au niveau BAS. C'est d'ailleurs ce qu'indique la façon dont sont désignées les sorties, $\overline{O}_7$, $\overline{O}_6$, $\overline{O}_5$ etc.; la barre de complémentation indique des sorties vraies au niveau BAS.

Le code d'entrée est appliqué aux bornes A_2, A_1 et A_0, où A_2 est le bit de poids fort. Comme il a trois entrées et huit sorties, c'est un décodeur entrée trois voies, sortie huit voies ou, ce qui est équivalent, un décodeur 1 parmi 8.

Les entrées $\overline{E}_1$, $\overline{E}_2$ et E_3 sont des entrées de validation distinctes qui sont combinées dans une porte ET. Pour valider les portes NON-ET de sortie afin qu'elles indiquent le code d'entrée $A_2A_1A_0$ correspondant, il faut que la sortie de cette porte ET soit au niveau HAUT. C'est ce qui se produit seulement quand $\overline{E}_1 = \overline{E}_2 = 0$ et $E_3 = 1$. Autrement dit, $\overline{E}_1$ et $\overline{E}_2$ sont vraies au niveau BAS, E_3 est vraie au niveau HAUT, et ces trois entrées doivent être dans leur état vrai pour rendre actives les sorties du décodeur. Si au moins une de ces entrées validation est dans son état non vrai, la sortie de ET est au niveau BAS, ce qui impose aux sorties de toutes les NON-ET leur état inactif HAUT, et cela quel que soit le code d'entrée. D'ailleurs, la figure 9-3 b) résume tout ceci dans une table de vérité. Rappelons que « x » veut toujours dire « condition indifférente ».

Le symbole logique du 74LS138 vous est montré à la figure 9-3 c). Notez comment on représente les sorties vraies au niveau BAS et les entrées de validation. Même si la porte ET de validation est dessinée à l'extérieur, en réalité, elle fait partie des circuits internes de la puce.

EXEMPLE 9-1

Indiquez les états des sorties d'un 74LS138 pour chacune des conditions d'entrée que voici:

a) $E_3 = \overline{E}_2 = 1, \overline{E}_1 = 0, A_2 = A_1 = 1, A_0 = 0$

b) $E_3 = 1, \overline{E}_2 = \overline{E}_1 = 0, A_2 = 0, A_1 = A_0 = 1$

Solution

a) Lorsque $\overline{E}_2 = 1$, le décodeur est désactivé et toutes ses sorties se trouvent dans leur état HAUT (inactif). C'est ce qu'on peut déduire de la table de vérité ou bien en relevant les niveaux logiques tout au long du circuit logique.

b) Toutes les sorties de validation sont activées, de sorte que la section de décodage est validée. Elle décode donc le code d'entrée $011_2 = 3_{10}$ afin d'activer la sortie $\overline{O}_3$. Donc, $\overline{O}_3$ se trouve au niveau BAS et toutes les autres sorties sont au niveau HAUT.

EXEMPLE 9-2

La figure 9-4 montre comment il est possible d'agencer quatre 74LS138 et un INVERSEUR pour obtenir un décodeur 1 parmi 32. Ces décodeurs sont notés Z1–Z4 afin de s'y référer facilement et les huit sorties de chacun de ceux-ci sont combinées pour donner un total de 32 sorties. Les sorties de Z1 sont $\overline{O}_0$–$\overline{O}_7$, celles de Z2 sont renommées $\overline{O}_8$–$\overline{O}_{15}$, celles de Z3 et de Z4 sont renommées respectivement $\overline{O}_{16}$–$\overline{O}_{23}$ et $\overline{O}_{24}$–$\overline{O}_{31}$. Un code d'entrée de 5 bits $A_4A_3A_2A_1A_0$ n'ouvre qu'une seule des 32 sorties pour chacune des 32 représentations d'entrée possibles.

a) Dites quelle sortie est active si $A_4A_3A_2A_1A_0 = 01101$.

b) Indiquez la plage des codes d'entrée qui rend actives toutes les sorties de la puce Z4.

Solution

a) Le code d'entrée à 5 bits est composé de 2 parties distinctes. Les bits A_4 et A_3 déterminent lequel des quatre boîtiers Z1–Z4 est validé, alors que la section $A_2A_1A_0$ fixe la sortie de la puce validée qui passe à son niveau vrai. Si $A_4A_3 = 01$, seul Z2 a ses entrées

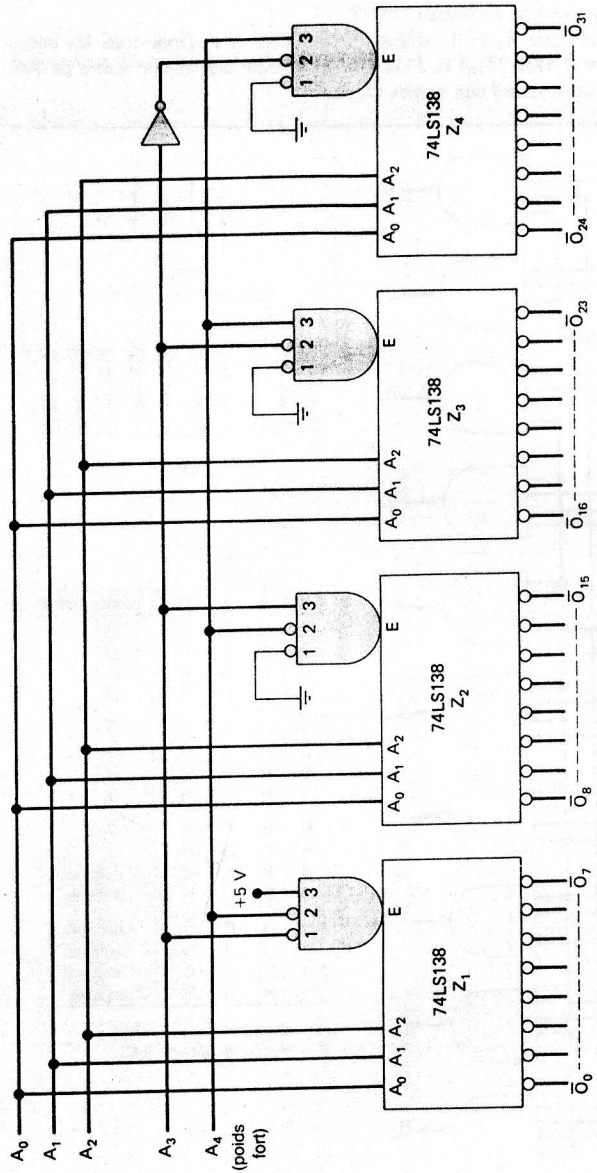
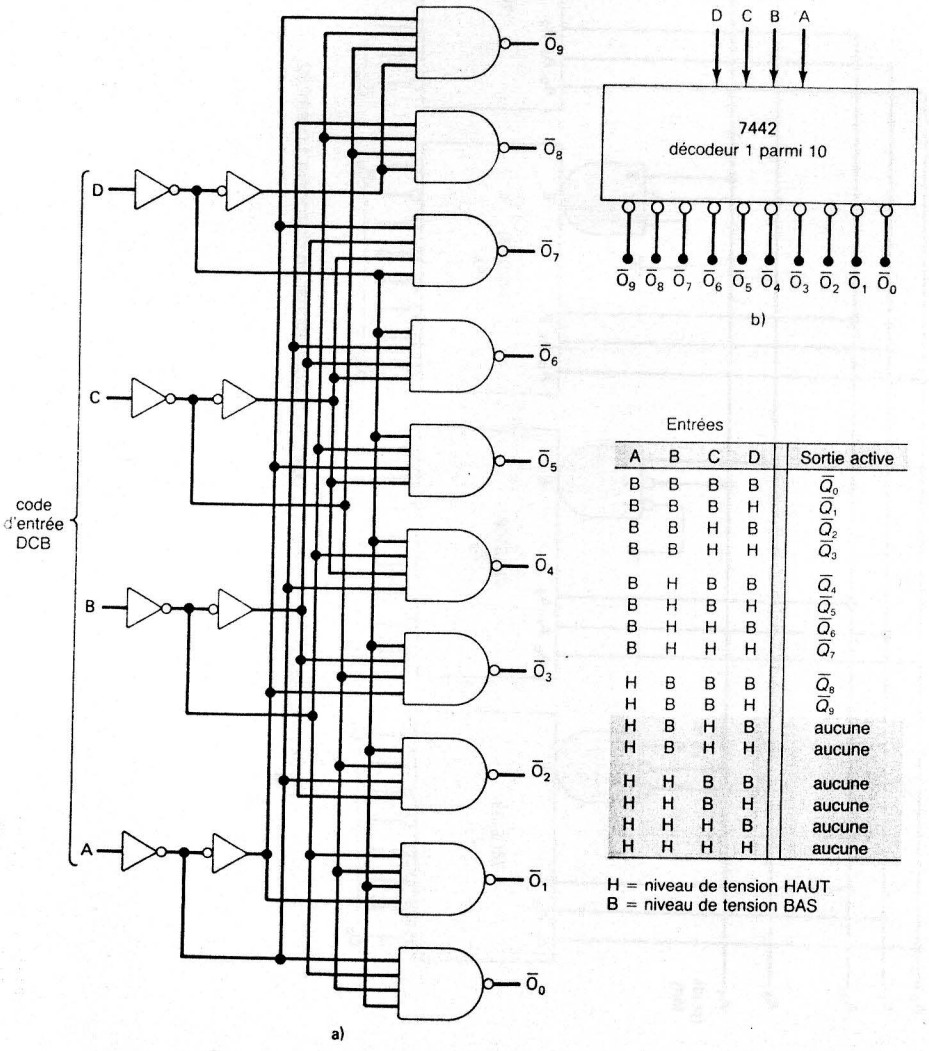


FIGURE 9-4 Quatre puces 74LS138 agencées pour donner un décodeur 1 parmi 32.

de validation au niveau vrai. Donc Z2 réagit au code $A_2A_1A_0 = 101$ et fait passer sa sortie $\overline{O}_5$ à sa valeur vraie, sortie rebaptisée $\overline{O}_{13}$. Donc le code d'entrée 01101, l'équivalent binaire du chiffre décimal 13, fait passer la sortie $\overline{O}_{13}$ au niveau BAS tandis que toutes les autres sorties restent au niveau HAUT.

b) Pour valider Z4, il faut que A_4 et A_3 soient tous les deux à 1. Donc tous les codes d'entrée compris entre $11000(24_{10})$ et $11111(31_{10})$ rendent active une sortie de Z4. Cette plage de codes correspond aux sorties $\overline{O}_{24}$ à $\overline{O}_{31}$.



Entrées

A	B	C	D	Sortie active
B	B	B	B	$\overline{Q}_0$
B	B	B	H	$\overline{Q}_1$
B	B	H	B	$\overline{Q}_2$
B	B	H	H	$\overline{Q}_3$
B	H	B	B	$\overline{Q}_4$
B	H	B	H	$\overline{Q}_5$
B	H	H	B	$\overline{Q}_6$
B	H	H	H	$\overline{Q}_7$
H	B	B	B	$\overline{Q}_8$
H	B	B	H	$\overline{Q}_9$
H	B	H	B	aucune
H	B	H	H	aucune
H	H	B	B	aucune
H	H	B	H	aucune
H	H	H	B	aucune
H	H	H	H	aucune

H = niveau de tension HAUT
B = niveau de tension BAS

FIGURE 9-5 a) Schéma logique du décodeur DCB-décimal 7442; b) son symbole logique; c) sa table de vérité. (Gracieuseté de Fairchild, une filiale Schlumberger).

Décodeurs DCB-décimal La figure 9-5 a) montre le schéma logique d'un *décodeur DCB-décimal 7442*, que l'on retrouve aussi en version 74LS42 et 74HC42. Une sortie ne passe à 0 qu'au moment où son entrée correspondante DCB est appliquée. Par exemple, la sortie $\overline{O}_5$ ne devient au niveau BAS que lorsque les valeurs sur les entrées sont $DCBA = 0101$; de même, $\overline{O}_8$ ne passe au niveau BAS que lorsque $DCBA = 1000$. Dans le cas des combinaisons DCB non valides, aucune des entrées n'est validée. Ce décodeur reçoit fréquemment le nom de *décodeur entrée 4 voies, sortie 10 voies* ou *décodeur 1 parmi 10*. Le symbole logique et la table de vérité du 7442 sont reproduits également sur cette figure. Remarquez l'absence dans ce décodeur d'une entrée de validation. Dans le problème 9-17, vous apprendrez comment on peut transformer un 7442 en un décodeur entrée 3 voies, sortie 8 voies (ou 1 parmi 8), dans lequel l'entrée D sert d'entrée de validation.

Décodeur/pilote DCB-décimal Le TTL 7445 est un *décodeur/pilote* DCB-décimal. Le terme « pilote » est ajouté à la désignation de cet élément pour indiquer des sorties en collecteur ouvert qui permettent à une sortie TTL de fonctionner avec un courant plus intense et dans des limites de tension supérieures aux valeurs normales. Les sorties du 7445 peuvent absorber jusqu'à 80 mA dans l'état BAS et être portées jusqu'à 30 V dans l'état HAUT. Ceci lui permet d'attaquer directement des charges comme des voyants DEL, des relais ou des moteurs à courant continu.

Applications du décodeur Les décodeurs servent chaque fois qu'une sortie ou un groupe de sorties ne doit être validé qu'à la réalisation d'une certaine combinaison des niveaux d'entrée. Ces niveaux d'entrée sont fréquemment fournis par un compteur ou un registre. Quand les entrées d'un décodeur sont alimentées par un compteur qui dénombre sans arrêt, les sorties du décodeur sont activées les unes à la suite des autres et peuvent alors servir à une tâche de synchronisation ou de séquencement pour mettre en marche ou à l'arrêt différents dispositifs. On a un exemple de ceci à la figure 9-6 où un compteur 74LS293 est couplé au décodeur/pilote 7445 déjà décrit.

EXEMPLE 9-3

Décrivez le fonctionnement du circuit de la figure 9-6 a).

Solution Le compteur est déclenché par un signal d'une impulsion par seconde, de sorte que la séquence qui apparaît en sortie est la suite binaire des nombres (rythme 1 nombre à la seconde). Les sorties des bascules du compteur sont raccordées aux entrées du décodeur. Les sorties $\overline{O}_3$ et $\overline{O}_6$ ont comme tâche d'ouvrir et de fermer les relais K_1 et K_2. Par exemple, quand $\overline{O}_3$ n'est pas dans son état vrai (niveau HAUT), son transistor de sortie est bloqué (ne conduit pas) et le courant ne peut circuler à travers le relais qui se trouve ainsi dans l'état au repos. Quand $\overline{O}_3$ est dans son état vrai (BAS), son transistor de sortie conduit et se comporte comme un absorbeur pour le courant qui traverse K_1, ce qui a pour effet d'alimenter K_1 (de le mettre en position de travail). Remarquez que les relais fonctionnent avec une alimentation de +24 V. En outre, remarquez la présence de diodes en parallèle avec les relais; ces dernières ont pour rôle de protéger les transistors de sortie du décodeur contre la tension de « contre-réaction inductive » qui peut se produire si le courant dans la bobine est stoppé brutalement.

Le chronogramme de la figure 9-6 b) illustre une séquence d'événements. Si on suppose que le compteur est à l'état 0000 à l'instant 0, les deux sorties $\overline{O}_3$ et $\overline{O}_6$ sont initialement dans leur état désactivé (état HAUT), les transistors sont bloqués et les deux relais sont au repos. À l'instant du FD de la troisième impulsion (instant 3), le compteur passe dans l'état 0011 (3). Cela a pour effet d'activer la sortie O_3 et par conséquent

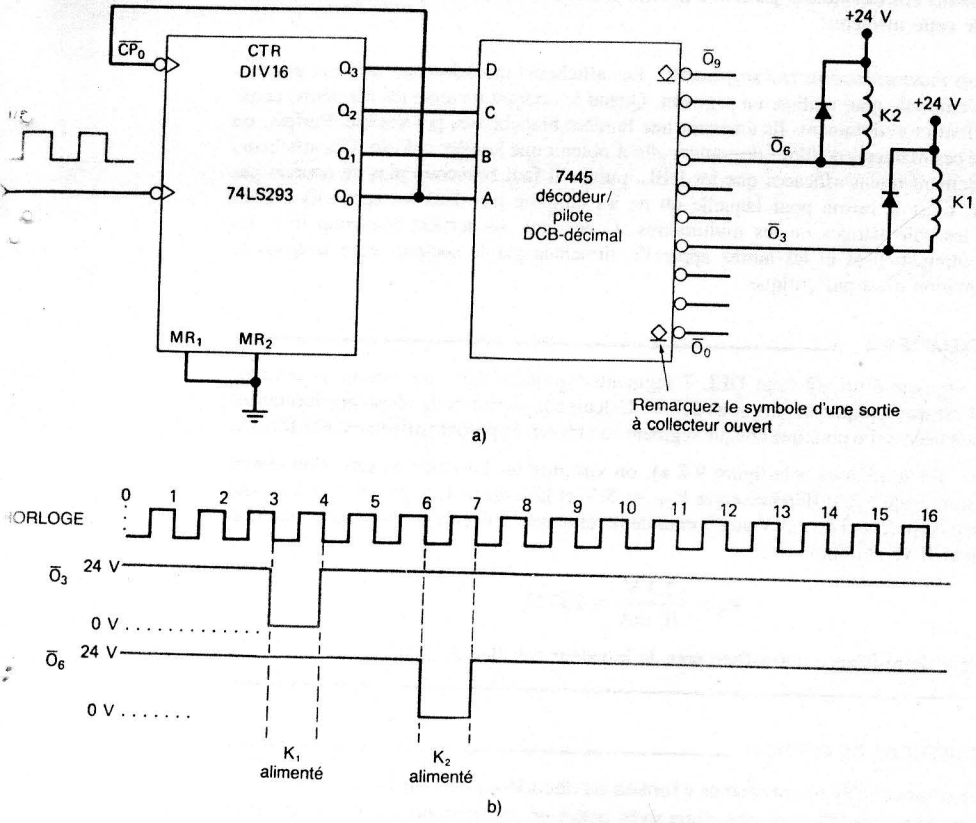


FIGURE 9-6 Exemple 9-3; le couple compteur/décodeur utilisé pour réaliser des tâches de synchronisation et de séquencement.

d'alimenter K_1. Au moment du FD de la quatrième impulsion, le compteur passe au nombre 0100 (4), ce qui désactive la sortie O_3, et remet le relais K_1 au repos.

Le même scénario se produit à l'instant 6 quand le compteur passe à l'état 0110 (6); la sortie $\overline{O}_6$ passe ainsi à 0 et alimente K_2. À l'instant 7, le compteur passe au nombre 0111 (7) et la sortie O_6 s'invalide et place K_2 au repos.

Le compteur continue de dénombrer les impulsions qui arrivent. Après 16 impulsions, la séquence recommence.

Les décodeurs sont largement utilisés dans les mémoires des ordinateurs: ce sont eux qui reçoivent le code d'adresse du processeur central et activent l'emplacement mémoire désigné par l'adresse. C'est une application que nous étudierons plus en détail au chapitre 11, quand nous aborderons le sujet des mémoires.

Un autre domaine d'applications considérable des décodeurs est celui de la conversion de données binaires en une forme se prêtant à un affichage numérique. Ce genre d'applications est justement le sujet des prochaines sections.

QUESTIONS DE RÉVISION

1. Est-ce qu'un décodeur peut avoir plus d'une de ses sorties activées en même temps?
2. Quelle est la fonction d'une entrée de validation dans un décodeur?
3. Qu'est-ce qui différencie un 7445 d'un 7442?
4. Le 74154 est un décodeur entrée 4 voies, sortie 16 voies disposant de deux entrées de validation vraies au niveau BAS. Combien de broches (en tenant compte de l'alimentation et de la masse) retrouve-t-on dans ce boîtier?

9-2 PILOTES/DÉCODEURS DCB-7 SEGMENTS

Dans de nombreux affichages numériques, les dix chiffres 0 à 9, et parfois les caractères hexadécimaux A à F, sont configurés au moyen de 7 segments [figure 9-7 a)]. Chaque segment est constitué d'un matériau qui émet de la lumière quand il est traversé par un courant. Les matériaux les plus utilisés sont les diodes électroluminescentes (DEL) et les filaments incandescents. La figure 9-7 b) nous montre comment sont disposés les segments afin de pouvoir afficher les différents chiffres. Par exemple, pour afficher le chiffre « 6 » il faut que les segments *c*, *d*, *e*, *f* et *g* soient allumés et que les segments *a* et *b* soient éteints.

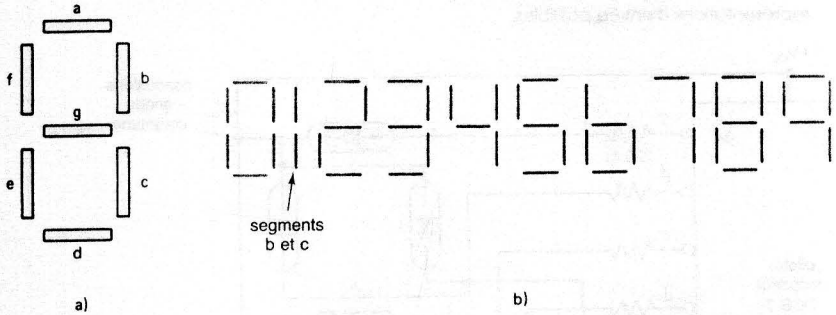


FIGURE 9-7 a) Disposition des 7 segments; b) les segments qui doivent être allumés pour former chacun des chiffres.

Un *pilote/décodeur DCB-7 segments* accepte en entrée les 4 bits DCB et rend actives les sorties qui vont permettre de faire passer un courant dans les segments qui forment le chiffre décimal correspondant. La logique de ce décodeur est plus compliquée que ce que l'on a vu précédemment, parce que chaque sortie peut être mise au niveau vrai dans plus d'une combinaison de bits d'entrée. Par exemple, le segment *e* est allumé lorsque sont formés les chiffres 0, 2, 6 et 8, c'est-à-dire quand les codes d'entrée sont 0000, 0010, 0110 ou 1000.

Le pilote/décodeur DCB-7 segments, illustré à la figure 9-8 a) (TTL 7446 ou 7447), pilote un afficheur DEL à 7 segments. Chaque segment est constitué d'une ou deux diodes électroluminescentes. Les anodes de ces diodes sont toutes réunies à V_{CC} (+5 V). Leurs cathodes sont connectées au travers de résistances limitatrices de courant aux sorties appropriées du pilote/décodeur. Celui-ci a des sorties qui sont vraies au niveau BAS, soit des transistors pilotes à collecteur ouvert pouvant absorber un courant passablement intense.

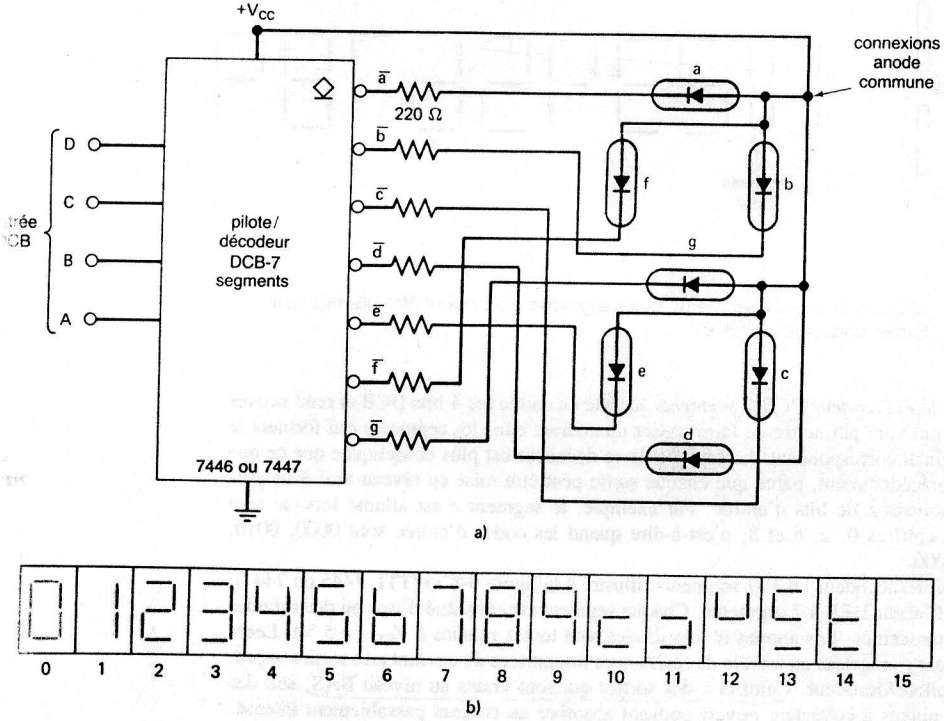
C'est la raison pour laquelle les afficheurs DEL peuvent nécessiter entre 10 mA et 40 mA par segment, selon leur type et leur taille.

Pour illustrer le fonctionnement de ce circuit, soit l'entrée DCB $D = 0$, $C = 1$, $B = 0$, $A = 1$, correspondant à la représentation DCB de 5. En réponse à cette entrée, les sorties $\overline{a}$, $\overline{f}$, $\overline{g}$, $\overline{c}$ et $\overline{d}$ du pilote/décodeur sont amenées au niveau BAS (raccordées à la masse), ce qui a pour effet de faire passer un courant à travers les segments DEL *a*, *f*, *g*, *c* et *d* et d'afficher le nombre 5. Les sorties $\overline{b}$ et $\overline{e}$ demeurent au niveau HAUT (ouvert) et les segments correspondants *b* et *e* restent éteints.

Les pilotes/décodeurs 7446 et 7447 ont été conçus de façon à allumer certains segments même si le code d'entrée est supérieur à 1001 (9). D'ailleurs à la figure 9-8 b), on peut voir les segments qui s'allument en réponse à tous les codes d'entrée de 0000 à 1111 (15). Notez que le code d'entrée 1111 éteint tous les segments.

L'afficheur DEL de la figure 9-8 est dit *à anode commune* parce que toutes les anodes des segments sont réunies à V_{CC}. Il existe un autre type d'afficheur DEL à 7 segments dit *à cathode commune* dans lequel toutes les cathodes des segments sont réunies à la masse. Ce dernier type d'afficheur est attaqué par un pilote/décodeur DCB-7 segments dont les sorties sont vraies au niveau HAUT; ainsi quand des sorties sont rendues actives, les anodes

FIGURE 9-8 a) Pilote/décodeur DCB-7 segments attaquant un afficheur DEL à 7 segments à anode commune; b) caractères configurés par les segments pour toutes les représentations d'entrée possibles.



des segments correspondants passent à la tension HAUTE. Le pilote/décodeur 7448 fonctionne de cette manière.

Afficheurs incandescents à 7 segments Les afficheurs incandescents utilisent des filaments d'ampoule pour réaliser un segment. Quand le courant traverse les filaments, ceux-ci chauffent et s'illuminent. Ils émettent une lumière blanche très perceptible. Parfois, on recouvre ces filaments de filtres de couleur afin d'obtenir une lumière colorée. Ces afficheurs sont nettement moins efficaces que les DEL, puisqu'il faut beaucoup plus de courant par segment; c'est la raison pour laquelle on ne les emploie pas dans les appareils à piles comme les calculatrices ou les multimètres. Cependant, ils servent beaucoup dans les caisses enregistreuses et les autres appareils alimentés par le secteur, pour lesquels la consommation n'est pas critique.

EXEMPLE 9-4

Chaque segment d'un afficheur DEL 7 segments typique donne une luminosité normale quand il est traversé par 10 mA sous 2,7 V. Calculez la valeur de la résistance limitatrice de courant nécessaire pour que chaque segment soit traversé approximativement par 10 mA.

Solution En se référant à la figure 9-8 a), on voit que la résistance en série doit chuter une tension égale à la différence entre $V_{CC} = 5$ V et la tension du segment, 2,7 V. Cette différence de potentiel de 2,3 V aux bornes de la résistance doit être produite par un courant de 10 mA. Il vient donc:

$$R_S = \frac{2{,}3\text{ V}}{10\text{ mA}} = 230\ \Omega$$

Une valeur de résistance normalisée près de la valeur calculée est 220 Ω.

QUESTIONS DE RÉVISION

1. Quels segments DEL seront allumés si l'entrée du décodeur/pilote est 1001?
2. *Vrai ou faux:* Il peut y avoir plus d'une sortie active en même temps dans un décodeur/pilote DCB-7 segments.

9-3 AFFICHEURS À CRISTAUX LIQUIDES (ACL)

Théoriquement, les afficheurs à cristaux liquides fonctionnent sous une basse tension (typiquement 3 à 15 V valeur efficace), à basse fréquence (25 à 60 Hz) et consomment très peu de courant. Souvent, ces afficheurs possèdent 7 segments afin de figurer les chiffres (voir figure 9-9). La tension alternative nécessaire pour allumer un segment est appliquée entre ce dernier et la *plaque arrière*, commune à tous les segments. Le segment et la plaque arrière forment un condensateur qui tire très peu de courant tant que la fréquence reste faible. Généralement, elle ne doit quand même pas être inférieure à 25 Hz, sans quoi les segments pourraient clignoter.

Les ACL consomment moins de courant que les afficheurs DEL et sont très utilisés dans les dispositifs à pile comme les calculatrices et les montres. Un ACL n'émet pas d'énergie lumineuse comme une DEL; il faut donc les éclairer de l'extérieur.

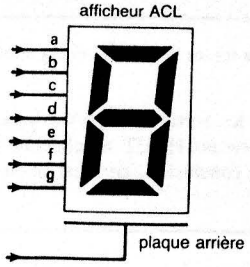


FIGURE 9-9 Afficheur à cristaux liquides.

Attaque d'un ACL Un segment à cristaux liquides est ALLUMÉ quand une tension alternative est appliquée entre lui et la plaque arrière, et est ÉTEINT quand il n'y a pas de tension entre lui et la plaque arrière. Il est de pratique courante de ne pas utiliser de signal alternatif, mais plutôt de générer la tension nécessaire en appliquant des ondes carrées en inversion de phase entre le segment et la plaque. C'est d'ailleurs ce que montre la figure 9-10 dans le cas d'un segment. Une onde carrée de 40 Hz est appliquée à la plaque arrière de même qu'à l'entrée d'une porte OU-EXCLUSIF CMOS 4070. À l'autre entrée de la porte, on applique un signal de COMMANDE qui donne l'ordre d'ALLUMER ou d'ÉTEINDRE le segment.

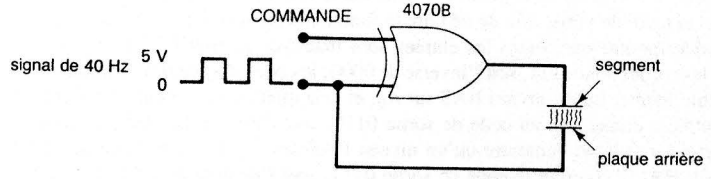


FIGURE 9-10 Méthode d'excitation d'un segment ACL. Quand COMMANDE est BASSE, le segment est ÉTEINT. Quand COMMANDE est HAUTE, le segment est ALLUMÉ.

Quand l'entrée COMMANDE est BASSE, la sortie de la porte OU-EXCLUSIF est une réplique de l'onde carrée de 40 Hz, de sorte que les signaux appliqués au segment et à la plaque arrière sont égaux. Il n'y a donc pas de différence de tension et le segment est ÉTEINT. Quand l'entrée COMMANDE est HAUTE, la sortie de la porte est l'INVERSE de l'onde carrée de 40 Hz et le signal appliqué au segment est en inversion de phase par rapport au signal appliqué à la plaque. Par conséquent, la tension du segment sera alternativement à $+5$ V et à -5 V par rapport à la plaque. Cette tension alternative ALLUME le segment.

Ce principe de base peut être appliqué à une rangée complète d'afficheurs ACL 7-segments, comme nous le montre la figure 9-11. Sur celle-ci, le pilote/décodeur DCB-7 segments CMOS 4511 fournit les signaux COMMANDE aux 7 portes OU-EXCLUSIF qui excitent les 7 segments. Les sorties du 4511 sont vraies au niveau HAUT, puisqu'il faut un niveau HAUT pour ALLUMER un segment.

Généralement, on a recours à des dispositifs CMOS pour piloter des ACL et cela pour deux raisons: 1) ils dissipent moins de puissance que les TTL et conviennent mieux

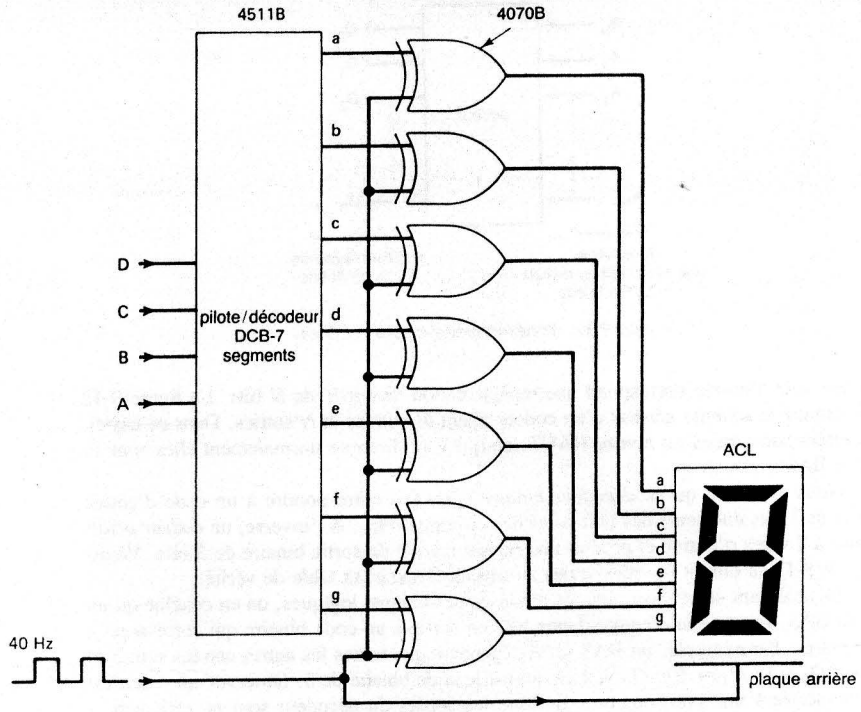


FIGURE 9-11 Excitation d'un ACL 7-segments.

aux applications à pile qui utilisent des afficheurs à cristaux liquides; 2) la tension de l'état BAS d'un TTL n'est pas précisément 0 V et peut même aller jusqu'à 0,4 V. Il y a donc entre le segment et la plaque arrière une composante de tension continue qui raccourcit considérablement la durée de vie d'un ACL.

QUESTION DE RÉVISION

1. Dites lesquelles des affirmations suivantes concernent les afficheurs ACL et lesquelles concernent les afficheurs DEL: a) émettent de la lumière; b) réfléchissent la lumière ambiante; c) sont idéaux pour les applications à faible consommation; d) exigent une tension alternative; e) dessinent les chiffres en exploitant un montage à 7 segments; f) ont besoin de résistances limitatrices de courant.

9-4 CODEURS

Si le décodage est un processus à partir duquel une représentation de N bits produit un signal HAUT (ou BAS) sur *une et seulement une* des lignes de sortie d'un décodeur, alors le processus inverse est le *codage* et il utilise un circuit logique appelé « codeur ». Un codeur a un certain nombre de voies d'entrée, dont une seule est active à la fois; à une

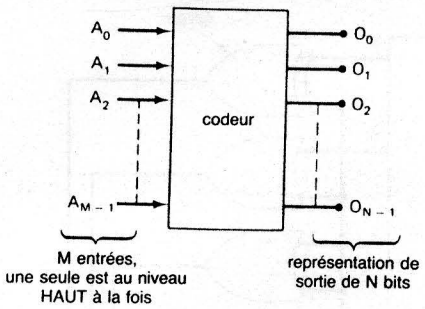


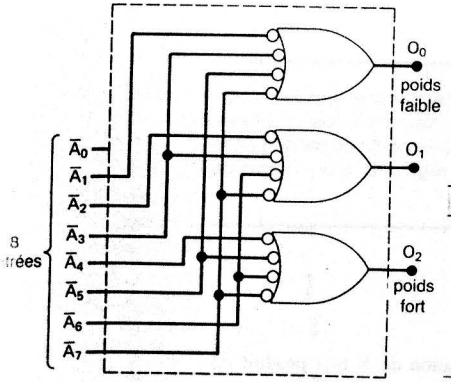
FIGURE 9-12 Schéma général d'un codeur.

certaine voie d'entrée correspond une représentation de *sortie* de N bits. La figure 9-12 nous montre le schéma général d'un codeur ayant M entrées et N sorties. Dans ce cas-ci, les entrées sont vraies au niveau HAUT, ce qui veut dire que normalement elles sont au niveau BAS.

Nous avons vu qu'un *décodeur binaire-octal* fait correspondre à un code d'entrée binaire de 3 bits une seule des huit voies de sortie possibles. À l'inverse, un *codeur octal-binaire* a 8 voies d'entrée et produit une représentation de sortie binaire de 3 bits. Voyez la figure 9-13 ce circuit (entrées vraies au niveau BAS) et sa table de vérité.

En analysant successivement les réponses des circuits logiques, on en conclue qu'un niveau BAS sur une seule entrée donne lieu en sortie à un code binaire qui correspond à cette entrée. Par exemple, un BAS sur $\overline{A}_3$ (pendant que toutes les autres entrées sont à 1) produit $O_2 = 0$, $O_1 = 1$ et $O_0 = 1$ ce qui est le code binaire de 3. Observez que $\overline{A}_0$ n'est pas connectée à une porte logique, puisque les sorties du décodeur sont normalement à 000 quand aucune des entrées de $\overline{A}_1$ à $\overline{A}_7$ n'est au niveau BAS.

FIGURE 9-13 Schéma logique d'un codeur octal-binaire (entrée 8 voies, sortie 3 voies). Ce circuit ne fonctionne correctement que lorsqu'une seule entrée est active à la fois.



Entrées								Sorties		
$\overline{A}_0$	$\overline{A}_1$	$\overline{A}_2$	$\overline{A}_3$	$\overline{A}_4$	$\overline{A}_5$	$\overline{A}_6$	$\overline{A}_7$	O_2	O_1	O_0
X	1	1	1	1	1	1	1	0	0	0
X	0	1	1	1	1	1	1	0	0	1
X	1	0	1	1	1	1	1	0	1	0
X	1	1	0	1	1	1	1	0	1	1
X	1	1	1	0	1	1	1	1	0	0
X	1	1	1	1	0	1	1	1	0	1
X	1	1	1	1	1	0	1	1	1	0
X	1	1	1	1	1	1	0	1	1	1

EXEMPLE 9-5

Déterminez les sorties du codeur de la figure 9-13 si $\overline{A}_3$ et $\overline{A}_5$ sont simultanément au niveau BAS.

Solution Analysez successivement les sorties des portes logiques et constatez que ces deux niveaux BAS provoquent des niveaux HAUT sur chacune des sorties; autrement dit, ils donnent le code binaire 111. En conclusion, ce n'est nullement le code de l'une ou l'autre des entrées activées.

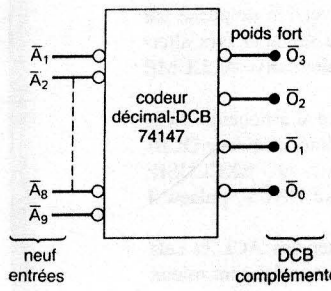
Codeurs de priorité Comme on l'a dit précédemment, le codeur de la figure 9-13 produit des résultats erronés si au moins deux entrées sont rendues actives simultanément. Un *codeur de priorité* est une version modifiée du codeur élémentaire et cette version modifiée possède les circuits logiques nécessaires pour que le code de sortie choisi, quand deux entrées sont actives, soit celui qui correspond au nombre le plus haut. Dans un tel codeur, quand les deux entrées $\overline{A}_5$ et $\overline{A}_3$ sont actives en même temps, la réponse donnée en sortie est 101. De même, ce codeur produit en sortie le code 110 si les entrées $\overline{A}_0$, $\overline{A}_2$ et $\overline{A}_6$ sont rendues actives en même temps. Les 74148, 74LS148 et 74HC148 sont des codeurs de priorité octal-binaire.

Codeur de priorité décimal-DCB 74147 La figure 9-14 nous montre le symbole logique du codeur de priorité décimal-DCB 74147 (74LS147, 74HC147). Celui-ci a 9 entrées vraies au niveau BAS représentant les 9 chiffres décimaux 1 à 9 et il produit la représentation DCB *complémentée* correspondant à l'entrée la plus haute mise au niveau vrai.

Examinons la table de vérité afin de déduire le fonctionnement de ce CI. La première ligne de cette table montre que toutes les entrées sont inactives au niveau HAUT. Pour cette condition, les sorties sont 1111, soit l'inverse de 0000, le code DCB de 0. La deuxième ligne de cette table montre qu'un niveau BAS sur $\overline{A}_9$, et cela quelles que soient les valeurs sur les autres entrées, donne lieu au code de sortie 0110, soit l'inverse de 1001, le code DCB de 9. La troisième ligne démontre qu'un niveau BAS sur $\overline{A}_8$, à la condition que le niveau sur $\overline{A}_9$ soit HAUT, produit le code de sortie 0111, soit l'inverse de 1000, le code DCB pour 8. Le même raisonnement pour toutes les lignes de cette table nous apprend qu'une entrée au niveau BAS, si toutes les entrées de rang supérieur sont au niveau HAUT, donne l'inverse du code DCB correspondant à cette entrée.

Les sorties du 74147 sont normalement à 1 quand aucune des entrées n'est à son niveau vrai. Ceci correspond à la condition d'entrée du chiffre décimal 0. Il n'y a en réalité

FIGURE 9-14 Codeur de priorité décimal-DCB 74147.



$\overline{A}_1$	$\overline{A}_2$	$\overline{A}_3$	$\overline{A}_4$	$\overline{A}_5$	$\overline{A}_6$	$\overline{A}_7$	$\overline{A}_8$	$\overline{A}_9$	$\overline{O}_3$	$\overline{O}_2$	$\overline{O}_1$	$\overline{O}_0$
1	1	1	1	1	1	1	1	1	1	1	1	1
X	X	X	X	X	X	X	X	0	0	1	1	0
X	X	X	X	X	X	X	0	1	0	1	1	1
X	X	X	X	X	X	0	1	1	1	0	0	0
X	X	X	X	X	0	1	1	1	1	0	0	1
X	X	X	X	0	1	1	1	1	1	0	1	0
X	X	X	0	1	1	1	1	1	1	0	1	1
X	X	0	1	1	1	1	1	1	1	1	0	0
X	0	1	1	1	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	1	1	0

X = 0 ou 1

aucune entrée $\overline{A}_0$, puisque le codeur suppose que l'état d'entrée du chiffre décimal 0 est celui créé quand toutes les autres entrées sont à 1. Pour obtenir le code DCB naturel à partir des sorties DCB complémentées du 74147, il faut ajouter un INVERSEUR à chacune des sorties.

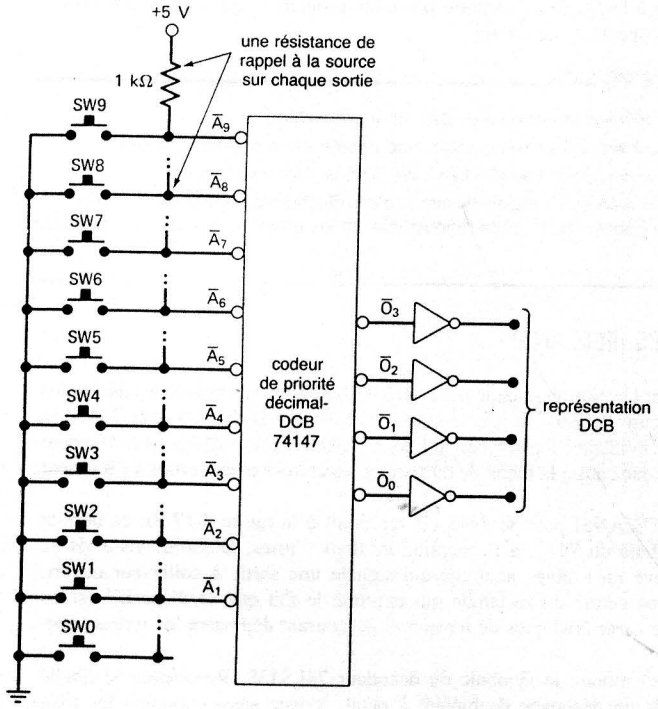
EXEMPLE 9-6

Déterminez les états des sorties de la figure 9-14 quand $\overline{A}_5$, $\overline{A}_7$ et $\overline{A}_3$ sont BASSES et que toutes les autres entrées sont HAUTES.

Solution La table de vérité nous apprend que lorsque $\overline{A}_7$ est BAS, les niveaux sur $\overline{A}_5$ et $\overline{A}_3$ n'ont pas d'importance. Pour cette raison, les sorties forment le code 1000.

Codeur d'interrupteurs À la figure 9-15, on montre comment on peut réaliser avec un 74147 un *codeur d'interrupteurs*. Les 10 interrupteurs peuvent être ceux d'un clavier de calculatrice représentant les chiffres 0 à 9. Les interrupteurs sont du type ouvert au repos, de sorte que les entrées du codeur sont généralement toutes à 1; quand c'est le cas la sortie DCB est 000 (remarquez les inverseurs). Quand une touche numérique est enfoncée, le circuit donne en sortie le code DCB correspondant à ce chiffre. Comme le 74147 est un

FIGURE 9-15 Codeur d'interrupteurs décimal-DCB.



codeur de priorité, l'enfoncement simultané de plusieurs touches place en sortie le code DCB de la touche numériquement la plus élevée.

Le codeur d'interrupteurs de la figure 9-15 peut servir toutes les fois que l'on doit introduire manuellement des donnés DCB dans un système numérique. Un exemple parfait de cela est la calculatrice électronique dans laquelle un opérateur introduit le nombre décimal en appuyant successivement sur des interrupteurs du clavier. Dans une calculatrice élémentaire, le code DCB de chaque chiffre décimal est introduit dans un registre mémoire de 4 bits. Ce qui veut dire que lorsqu'on appuie sur la première touche, le code DCB correspondant à ce chiffre est envoyé à un registre de 4 bits, quand on appuie sur le second interrupteur, le code DCB de ce nouveau chiffre est envoyé à un *autre* registre à bascules de 4 bits, et ainsi de suite. Une calculatrice qui peut traiter des nombres de 8 chiffres possède 8 registres de 4 bits pour mémoriser les codes DCB de ces chiffres. Les sorties de chacun de ces registres sont délivrées à un pilote/décodeur et à un affichage numérique, afin de présenter sur l'écran le nombre de 8 chiffres.

L'opération que l'on vient juste de décrire est réalisable par le circuit de la figure 9-16. Ce circuit accepte 3 chiffres décimaux introduits dans l'ordre à partir du clavier, code ces chiffres en DCB et mémorise la représentation DCB dans 3 registres de sortie à bascules. Les 12 bascules de type D Q_0–Q_{11} mémorisent les codes DCB de ces chiffres. Q_8–Q_{11} conservent le code DCB du chiffre de poids le plus fort, le premier introduit au clavier. Q_4–Q_7 conservent le chiffre de deuxième rang et Q_0–Q_3 conservent le chiffre de troisième rang. Les bascules X, Y et Z constituent un compteur circulaire (chapitre 7) qui commande le transfert des données des sorties du codeur au registre de sortie approprié. La porte OU voit apparaître sur sa sortie un niveau HAUT chaque fois qu'une des touches est enfoncée. Cette sortie peut être affectée par les rebonds des contacts de l'interrupteur et peut subir plusieurs oscillations avant de s'établir définitivement dans l'état HAUT. Le monostable sert à neutraliser les rebonds de l'interrupteur puisqu'il est déclenché par le premier front montant de la sortie de la porte OU et demeure dans un état HAUT pendant 20 ms, une durée nettement suffisante pour attendre que l'interrupteur ait cessé de rebondir. La sortie du monostable synchronise le compteur circulaire.

Voici le fonctionnement détaillé de ce circuit pour le cas où le nombre décimal introduit est 309:

1. On enfonce la touche REMISE À ZÉRO. Ce qui amène à 0 toutes les bascules mémoires Q_0–Q_{11}, ainsi que les bascules X et Y et met à 1 la bascule Z afin que l'état initial du compteur circulaire soit 001.
2. On lâche la touche REMISE À ZÉRO et on appuie sur la touche « 3 ». Les sorties du codeur 1100 sont complémentées pour donner 0011, le code DCB de 3. Ces valeurs binaires sont envoyées aux entrées D des trois registres de sortie de 4 bits.
3. La sortie de la porte OU passe au niveau HAUT (puisque deux de ses entrées sont au niveau HAUT) et met la sortie du monostable Q à 1 pendant 20 ms. Après cette durée, Q revient au niveau BAS et synchronise le compteur circulaire qui se retrouve alors dans l'état 100 (X passe à 1). Le front montant qui s'observe en X est délivré aux entrées *CLK* des bascules Q_8–Q_{11}, de sorte que les sorties du codeur sont transférées à ces bascules. C'est-à-dire $Q_{11} = 0$, $Q_{10} = 0$, $Q_9 = 1$ et $Q_8 = 1$. Remarquez que les bascules Q_0–Q_7 ne réagissent pas à la valeur placée sur leur entrée parce que leurs entrées *CLK* ne sont pas stimulées par un front montant.
4. La touche « 3 » est relâchée et la sortie de la porte OU revient au niveau BAS. La touche « 0 » est ensuite enfoncée. Celle-ci produit le code DCB 0000, code fourni aux entrées des bascules mémoires.

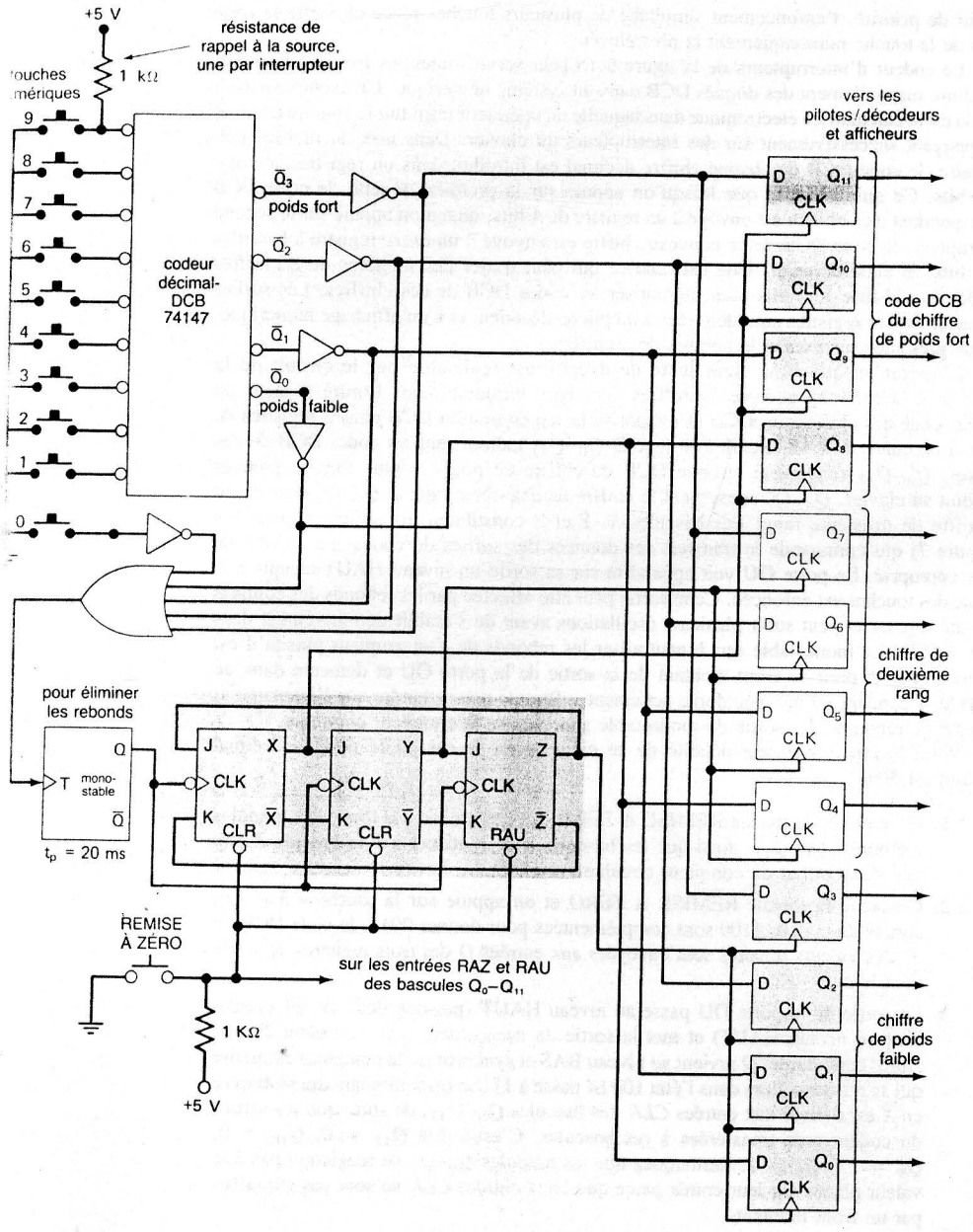


FIGURE 9-16 Circuit pour l'introduction à partir d'un clavier de nombres de 3 chiffres placés dans des registres mémoires.

5. La sortie de la porte OU passe à l'état HAUT en réponse à l'enfoncement de la touche « 0 » (remarquez l'inverseur) et déclenche le monostable pendant 20 ms. Après cette temporisation de 20 ms, le compteur circulaire passe dans l'état 010 (Y passe à 1). Le front montant présent sur la borne Y est communiqué aux entrées *CLK* de Q_4–Q_7 et transfère le 0000 à ces bascules. Notez encore une fois que les bascules Q_0–Q_3 et Q_8–Q_{11} ne réagissent pas à ce front montant sur Y.
6. La touche « 0 » est relâchée et la sortie de la porte OU revient au niveau BAS. C'est maintenant la touche « 9 » qui est enfoncée, donnant lieu aux sorties DCB 1001, délivrées aux bascules mémoires.
7. La sortie de la porte OU revient au niveau HAUT, déclenchant le monostable qui à son tour synchronise le compteur circulaire pour le remettre dans l'état 001 (Z passe au niveau HAUT). Le front montant qu'on observe sur la borne Z est communiqué aux entrées *CLK* de Q_0–Q_3 et transfère la représentation 1001 à ces dernières bascules. Les autres bascules mémoires ne réagissent pas.
8. À ce point-ci, on lit dans le registre mémoire à partir de Q_{11} 0011 0000 1001. Il s'agit de la représentation DCB du chiffre 309. Les sorties de ce registre alimentent des pilotes/décodeurs qui excitent les afficheurs appropriés afin de présenter le nombre décimal 309.
9. Les sorties des bascules mémoires sont également délivrées à d'autres circuits du système. Par exemple, dans une calculatrice, ces sorties sont envoyées à la section arithmétique en vue de participer à une opération.

Vous trouverez à la fin de ce chapitre plusieurs problèmes qui s'adressent particulièrement à d'autres aspects de ce circuit.

QUESTIONS DE RÉVISION

1. Dites quelle est la différence entre un codeur et un décodeur.
2. Dites quelle est la différence entre un codeur de priorité et un codeur ordinaire.
3. Quelles seront les sorties de la figure 9-15, si SW6, SW5 et SW2 sont fermés.
4. Décrivez le rôle des éléments suivants du circuit d'introduction au clavier de la figure 9-16: a) porte OU; b) le codeur 74147; c) le monostable; d) les bascules *X*, *Y*, *Z*; e) les bascules Q_0 à Q_{11}.

9-5 SYMBOLES IEEE/ANSI

Examinons maintenant la solution retenue par l'IEEE/ANSI pour symboliser les décodeurs et les codeurs. La figure 9-17 a) illustre le symbole IEEE/ANSI du décodeur 7442. Sa signification est assez évidente. Remarquez le libellé BCD/DEC qui désigne un décodeur BCD-décimal. Remarquez aussi la façon de numéroter les entrées et les sorties à l'intérieur du rectangle.

Le symbole IEEE/ANSI pour le 7445 est reproduit à la figure 9-17 b); ce dernier est identique au symbole du 7442, à l'exception de deux choses. D'abord, vis-à-vis de chaque sortie se trouve un losange souligné qui signale une sortie à collecteur ouvert. Ensuite, un triangle au centre du rectangle qui exprime le fait que ce dispositif est un tampon-pilote doté de caractéristiques de tension et de courant dépassant les valeurs normales.

La figure 9-17 c) montre le symbole du décodeur 74LS138. Remarquez le libellé BIN/OCT qui signifie un décodage de binaire à octal. Voyez aussi comment les trois

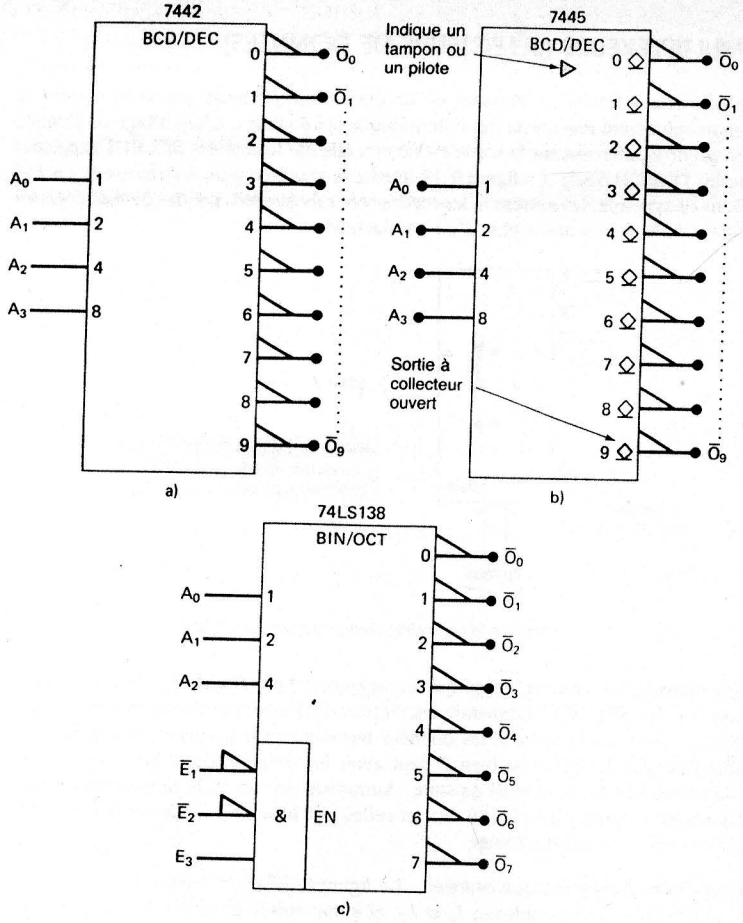


FIGURE 9-17 Symboles IEEE/ANSI de quelques décodeur.

entrées de validation sont combinées dans une porte ET afin de constituer un seul signal de validation EN.

La figure 9-18 contient, pour sa part, le symbole IEEE/ANSI du boîtier décodeur 74147. Le libellé HPRI/BCD nous apprend qu'il s'agit de la fonction de conversion de l'entrée active ayant la priorité la plus élevée en son code BCD. Remarquez de nouveau la numérotation des entrées et des sorties à l'intérieur du rectangle.

QUESTIONS DE RÉVISION

1. Que signifie le libellé BCD/DEC?
2. Que signifie le libellé HPRI?
3. Que signifie le signe ▷ à l'intérieur du rectangle d'un symbole.

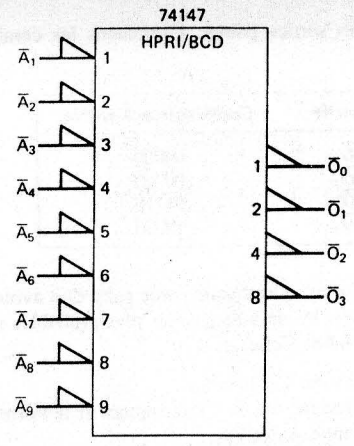


FIGURE 9-18 Symbole IEEE/ANSI du codeur 74147.

9-6 DÉPANNAGE

À mesure que se complexifient les circuits et les systèmes, le nombre de causes possibles pouvant produire des défaillances s'accroît dans les mêmes proportions. Même si la procédure de dépistage et de correction des défaillances demeure essentiellement identique, l'application de la technique **observation et analyse** revêt une importance encore plus grande pour le cas des circuits complexes, car le technicien parvient à cerner la panne à une petite zone du circuit. Cela ramène dans des proportions acceptables le nombre d'étapes des essais et le nombre de données à analyser. Grâce à sa compréhension du fonctionnement des circuits et à l'observation des symptômes de la panne, le technicien peut raisonner et souvent prédire les causes possibles du dysfonctionnement sans même avoir à toucher à une sonde ou à un oscilloscope. La technique de l'observation et de l'analyse n'est pas toujours mise en pratique par les techniciens inexpérimentés, sans doute à cause du nombre impressionnant d'appareils de mesure sophistiqués dont nous disposons. Il est toujours plus facile de se fier à ces outils et de laisser de côté le raisonnement et l'analyse.

Voici des exemples qui illustrent comment la technique de l'observation et de l'analyse peut être mise en application. En outre, vous trouverez à la fin de ce chapitre plusieurs problèmes qui s'efforcent de développer votre aptitude à recourir à cette technique.

EXEMPLE 9-7

Une technicienne teste le circuit de la figure 9-4 en utilisant un ensemble d'interrupteurs pour imposer le code d'entrée sur A_4 à A_0. Elle réalise toutes les combinaisons possibles du code d'entrée et vérifie la sortie du décodeur en vue de constater qu'il fonctionne. Elle observe que toutes les sorties impaires réagissent correctement, tandis que les sorties paires restent muettes quand on applique la combinaison d'entrée correspondante. Quelles sont les causes possibles de ce mauvais fonctionnement?

Solution Dans une situation de cette nature où un grand nombre de sorties sont fautives, il n'est pas réaliste de présager que chacune de ces sorties pourrait être défaillante. Il est plus probable qu'une condition d'entrée anormale soit à l'origine du problème. Quel est

le point commun à toutes les sorties paires. Examinons les combinaisons d'entrée de quelques-unes d'entre elles.

Sortie	Combinaison d'entrée
$\overline{O}_0$	00000
$\overline{O}_4$	00100
$\overline{O}_{14}$	01110
$\overline{O}_{18}$	10010

Ce qui se dégage tout de suite, c'est que chaque sortie paire doit avoir $A_0 = 0$ pour qu'elle devienne active. Par conséquent, les défaillances les plus probables sont celles qui empêchent A_0 de passer au niveau BAS. Celles-ci sont:

Un interrupteur défectueux sur l'entrée A_0
Une rupture dans le conducteur reliant l'interrupteur et la borne A_0
Un court-circuit externe entre A_0 et V_{CC}
Un court-circuit interne entre V_{CC} et les entrées A_0 de l'une des puces du décodeur

Donc, on est parvenu à délimiter la faute à une zone particulière du circuit. La défaillance exacte peut être retracée au moyen des techniques d'essai et de mesure que vous connaissez déjà.

EXEMPLE 9-8

Un technicien câble les sorties d'un compteur DCB avec les entrées du décodeur/pilote de la figure 9-8. Il délivre des impulsions au compteur à un rythme très lent et observe les afficheurs DEL qui produisent les nombres suivants quand le compteur progresse de 0000 à 1001. Examinez cette séquence et essayez de prévoir la cause la plus probable de cet affichage erroné.

NOMBRE	0	1	2	3	4	5	6	7	8	9
Affichage observé	0	1	0	3	d	b	6	7	8	a
Affichage prévu	0	1	2	3	4	5	6	7	8	9

Solution La comparaison de l'affichage observé et de l'affichage prévu permet de mettre en lumière les points importants que voici.

Pour les nombres dont l'affichage est incorrect, aucune des combinaisons de segments qui s'illuminent ne correspond au symbole d'un nombre supérieur à 1001. Cela écarte un compteur défectueux ou un mauvais câblage du compteur avec le décodeur/pilote.
Toutes les configurations de segments exactes (0, 1, 3, 6, 7 et 8) contiennent les segments *e* et *f*, tous les deux allumés ou tous les deux éteints.
Les configurations de segments erronées ont en commun les segments *e* et *f* dans des états opposés; si on intervertit ces deux segments, on obtient le bon chiffre.

La conclusion qui s'impose est que le technicien a probablement « croisé » les connexions vers les segments *e* et *f*.

9-7 MULTIPLEXEURS (SÉLECTEURS DE DONNÉES)

Un *multiplexeur* ou *sélecteur de données* est un circuit logique ayant plusieurs entrées de données, mais seulement *une* sortie qui communique les données. L'aiguillage de l'entrée de données qui nous intéresse sur la sortie est commandé par les entrées SELECT (appelées parfois entrées D'ADRESSE). La figure 9-19 illustre le symbole d'un multiplexeur général (MUX). Dans ce symbole, les entrées et les sorties sont représentées par des flèches épaisses afin d'indiquer qu'il peut y avoir plus d'un conducteur.

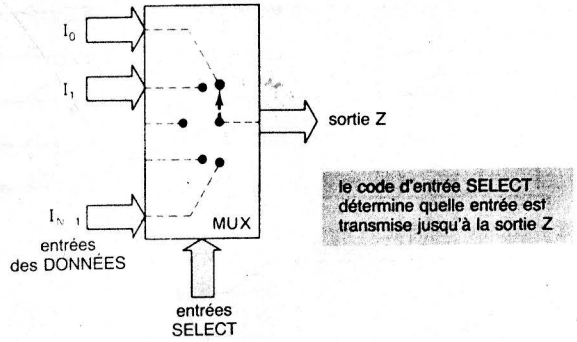


FIGURE 9-19 Symbole d'un multiplexeur numérique (MUX).

Un multiplexeur se comporte comme un commutateur* dans lequel un code numérique appliqué aux entrées SELECT commande les entrées de données qui sont raccordées à la sortie. Il peut y avoir sur la sortie Z les données introduites sur I_0 quand on applique un certain code d'entrée SELECT; ou bien Z peut avoir les données de I_1 en réponse à un autre code d'entrée SELECT; et ainsi de suite. Autrement dit, un multiplexeur choisit une source de données d'entrée parmi N et transmet celles-ci à la seule voie de sortie existante. C'est ce qu'on appelle le **multiplexage**.

Multiplexeur élémentaire à deux entrées La figure 9-20 schématise les circuits logiques d'un multiplexeur à deux entrées, I_0 et I_1, et à une entrée de sélection S. Le niveau logique appliqué à S choisit la porte ET qui est validée, de façon à ce que son entrée de données traverse la porte OU jusqu'à la sortie Z. D'après l'algèbre booléenne, l'expression de la sortie est:

$$Z = I_0\overline{S} + I_1S$$

Quand $S = 0$, cette expression se réduit à:

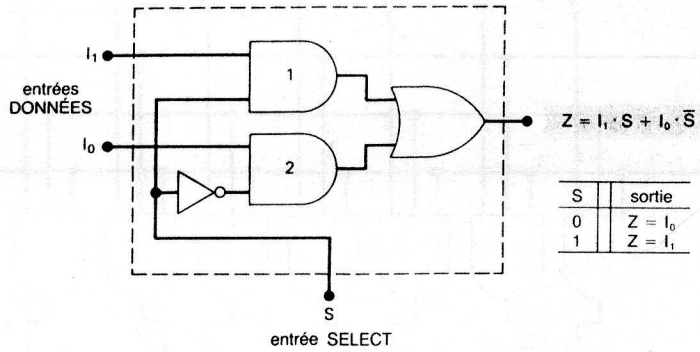
$$Z = I_0 \cdot 1 + I_1 \cdot 0$$
$$= I_0$$

ce qui indique que Z est identique au signal d'entrée I_0, signal qui peut être un niveau logique permanent ou un signal logique variable. Quand $S = 1$, l'expression après substitution devient

$$Z = I_0 \cdot 0 + I_1 \cdot 1 = I_1$$

ce qui montre que la sortie Z est dans ce cas identique à l'entrée I_1.

* Indiqué par les lignes en tiretées de la figure 9-19. Ces lignes ne font pas partie des circuits du MUX ni de son symbole et ont pour simple but d'illustrer la fonction de multiplexage.



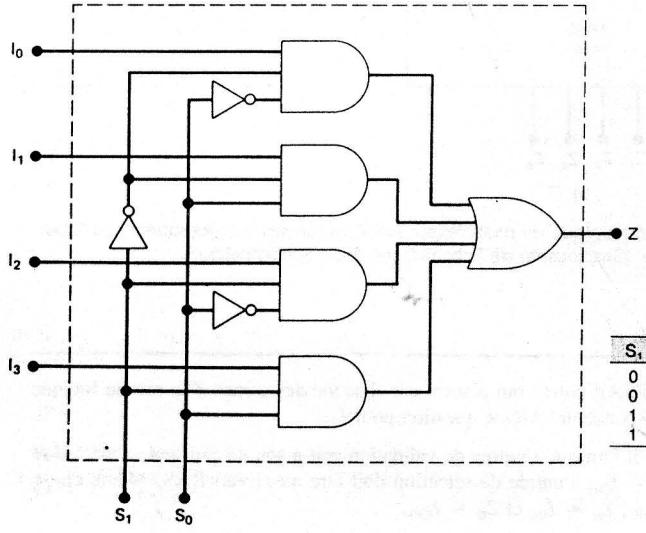
S	sortie
0	$Z = I_0$
1	$Z = I_1$

FIGURE 9-20 Multiplexeur à deux entrées.

Multiplexeur quatre entrées Le même principe de base sert à construire un multiplexeur à quatre entrées, comme celui de la figure 9-21. Dans ce cas-ci, il y a quatre entrées qui sont transmises à la sortie selon le choix indiqué par l'une des quatre combinaisons possibles des entrées de sélection S_1S_0. Chaque entrée de données aboutit à une porte ET dont les deux autres entrées correspondent à une certaine combinaison des niveaux des entrées de sélection; ainsi I_0 est réuni avec $\overline{S}_1\overline{S}_0$ de sorte que I_0 traverse la porte ET et atteint la sortie Z seulement quand $S_1 = 0$ et $S_0 = 0$. La table accompagnant cette figure donne la valeur de la sortie pour les trois autres codes de sélection d'entrée.

On peut se procurer dans le commerce des multiplexeurs à deux, quatre, huit et seize entrées fabriqués selon la technologie TTL ou CMOS. Ces CI de base peuvent être combinés afin de multiplexer un nombre d'entrées encore plus grand.

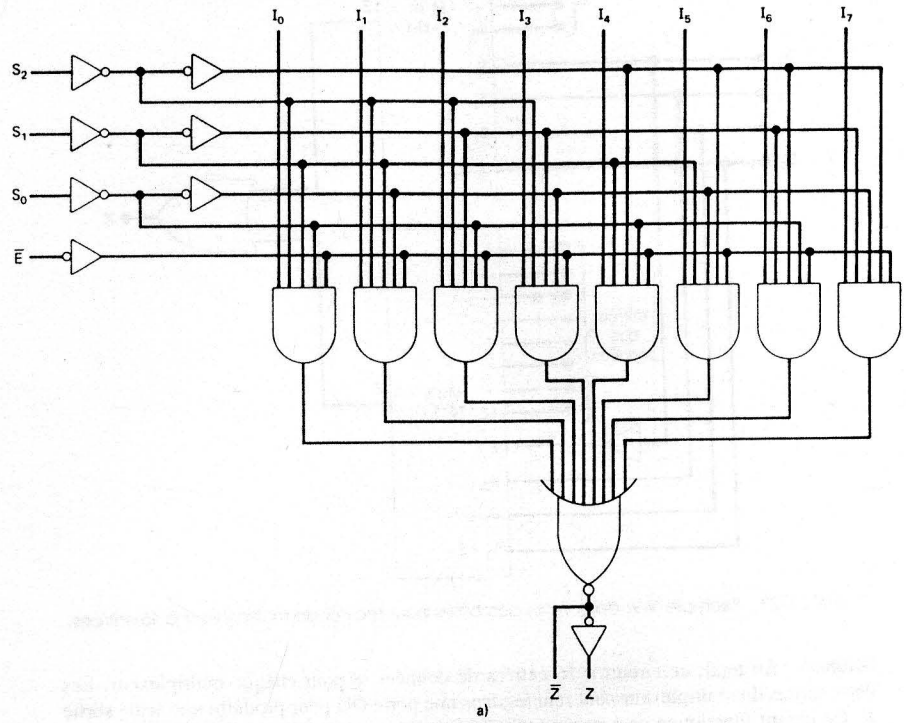
FIGURE 9-21 Multiplexeur à quatre entrées.



S_1	S_0	sortie
0	0	$Z = I_0$
0	1	$Z = I_1$
1	0	$Z = I_2$
1	1	$Z = I_3$

Multiplexeur à huit entrées Voyez à la figure 9-22 a) le schéma logique du multiplexeur à huit entrées 74151 (74LS151, 74HC151). Ce multiplexeur dispose d'une entrée validation, $\overline{E}$, et il fournit la sortie normale et la sortie complémentée. Quand $\overline{E} = 0$, les entrées de sélection $S_2S_1S_0$ choisissent une entrée de données (I_0 à I_7) dont les valeurs se retrouvent sur la sortie Z. Quand $\overline{E} = 1$, le multiplexeur est invalidé de sorte que $Z = 0$ quel que soit le code d'entrée de sélection. Ce fonctionnement est résumé dans le tableau de la figure 9-22 b); le symbole logique du 74151 est illustré à la figure 9-22 c).

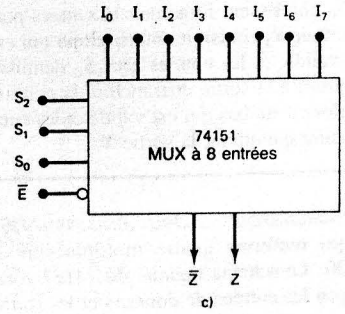
FIGURE 9-22 a) Schéma logique du multiplexeur 74151; b) sa table de vérité; c) son symbole logique. (Gracieuseté de Fairchild, filiale Schlumberger).



a)

entrées				sorties	
$\overline{E}$	S_2	S_1	S_0	$\overline{Z}$	Z
H	X	X	X	H	B
B	B	B	B	$\overline{I}_0$	I_0
B	B	B	H	$\overline{I}_1$	I_1
B	B	H	B	$\overline{I}_2$	I_2
B	B	H	H	$\overline{I}_3$	I_3
B	H	B	B	$\overline{I}_4$	I_4
B	H	B	H	$\overline{I}_5$	I_5
B	H	H	B	$\overline{I}_6$	I_6
B	H	H	H	$\overline{I}_7$	I_7

b)



c)

EXEMPLE 9-9

Le circuit de la figure 9-23 a recours à deux 74151, à un INVERSEUR et à une porte OU. Décrivez son fonctionnement.

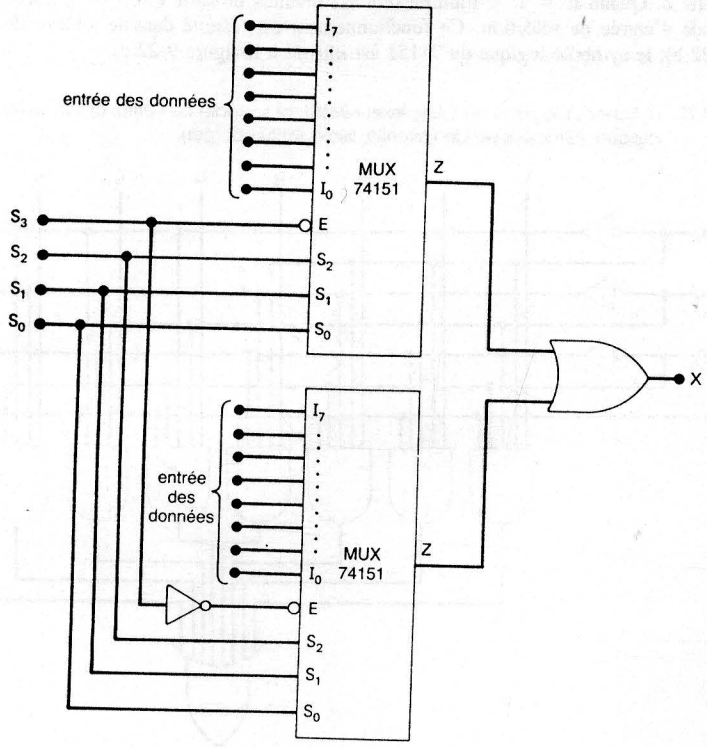
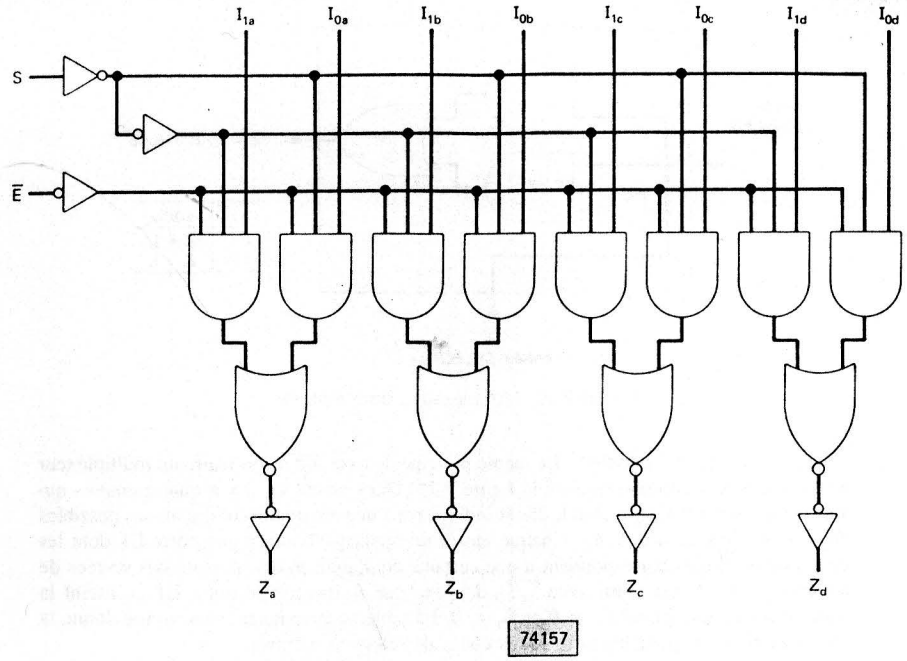


FIGURE 9-23 Exemple 9-9: deux 74151 combinés pour réaliser un multiplexeur à 16 entrées.

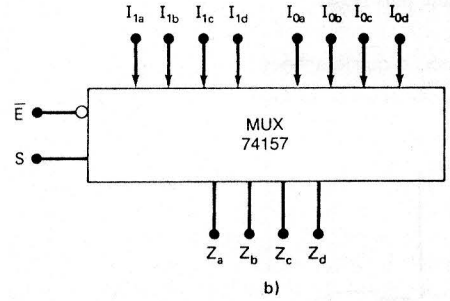
Solution Au total, ce circuit a 16 entrées de données, 8 pour chaque multiplexeur. Les deux sorties du multiplexeur sont réunies dans une porte OU pour produire une seule sortie X. Ce circuit fonctionne comme un multiplexeur à 16 entrées. Les 4 entrées de sélection $S_3S_2S_1S_0$ choisissent l'une des 16 entrées pour la faire passer jusqu'à X.

L'entrée S_3 choisit le multiplexeur qui est validé. Quand $S_3 = 0$, le multiplexeur du haut est validé, et les entrées $S_2S_1S_0$ définissent laquelle des entrées de ce multiplexeur passera jusqu'à la sortie et franchira la porte OU pour atteindre X. Quand $S_3 = 1$, c'est le multiplexeur du bas qui est validé et les entrées $S_2S_1S_0$ choisissent l'entrée de données de ce dernier qui atteint la sortie X.

Quatre multiplexeurs à deux entrées (74157/LS157/HC157) Il s'agit ici d'un boîtier intégré qui renferme quatre multiplexeurs à deux entrées identiques à celui de la figure 9-20. Le schéma logique du 74157 vous est montré à la figure 9-24 a). Notez de quelle façon les entrées de données et les sorties sont désignées.



a)



b)

$\overline{E}$	S	Z_a	Z_b	Z_c	Z_d
H	X	B	B	B	B
B	B	I_{0a}	I_{0b}	I_{0c}	I_{0d}
B	H	I_{1a}	I_{1b}	I_{1c}	I_{1d}

c)

FIGURE 9-24 a) Schéma logique du multiplexeur 74157; b) son symbole logique; c) sa table de vérité. (Gracieuseté de Fairchild, une filiale Schlumberger).

EXEMPLE 9-10

Déterminez les conditions d'entrée qui placent sur chacune des sorties Z le niveau logique de son entrée I_0 correspondante. Même question pour I_1.

Solution Avant tout, il faut que l'entrée de validation soit à son niveau vrai, c'est-à-dire $\overline{E} = 0$. Pour que $Z_a = I_{0a}$, l'entrée de sélection doit être au niveau BAS. Même chose pour produire $Z_b = I_{0b}$, $Z_c = I_{0c}$ et $Z_d = I_{0d}$.

Lorsque $\overline{E} = 0$ et $S = 1$, les sorties Z affichent les niveaux de leur entrée I_1 correspondante; c'est-à-dire $Z_a = I_{1a}$, $Z_b = I_{1b}$, $Z_c = I_{1c}$ et $Z_d = I_{1d}$.

Toutes les sorties sont invalidées (à 0) quand $\overline{E} = 1$.

Il peut être plus facile de considérer ce multiplexeur comme un simple multiplexeur à deux entrées, dont chacune des entrées aurait quatre voies et chacune des sorties quatre voies également. Les quatre voies de sortie passent alternativement d'un ensemble de quatre voies d'entrée à un autre selon la valeur placée sur l'entrée de sélection. Le symbole logique du 74157 est illustré à la figure 9-24 b).

QUESTIONS DE RÉVISION

1. Quelle est la fonction des entrées de sélection d'un multiplexeur?
2. Un multiplexeur peut diriger l'une de ses 32 données d'entrée vers sa sortie. Combien de bornes d'entrée différentes ce multiplexeur possède-t-il?

9-8 APPLICATIONS DES MULTIPLEXEURS

Les applications des multiplexeurs dans le domaine des techniques numériques sont nombreuses et variées. On en retrouve dans les circuits de sélection de données, d'aiguillage de données, d'ordonnancement des opérations, de conversion parallèle-série, de génération de formes d'ondes et de production de fonction logique. Nous nous proposons d'étudier quelques applications dans la présente section et de vous initier à certaines autres applications dans des problèmes en fin du chapitre.

Aiguillage des données Les multiplexeurs peuvent servir à aiguiller des données entre plusieurs sources et une seule destination. Une application typique est l'utilisation de multiplexeurs 74157 pour choisir et afficher le contenu d'un compteur DCB parmi 2 en utilisant une *seule* unité de pilotes/décodeurs et d'afficheurs DEL. Voyez le montage d'un tel circuit à la figure 9-25.

Chaque compteur est formé de deux étages DCB en cascade et chacun de ces compteurs est synchronisé par son propre signal d'horloge. Quand la ligne SÉLECTION DE COMPTEUR est HAUTE, les sorties du compteur 1 sont autorisées à passer à travers les multiplexeurs pour atteindre les pilotes/décodeurs et exciter les afficheurs DEL. Quand la ligne SÉLECTION DE COMPTEUR = 0, les sorties du compteur 2 traversent les multiplexeurs jusqu'aux afficheurs. De cette manière, le contenu d'un compteur ou de l'autre est affiché selon le signal envoyé par l'entrée SÉLECTION DE COMPTEUR. On retrouve couramment un tel circuit dans les montres numériques. Les circuits d'une montre numérique renferment plusieurs compteurs et registres qui marquent les secondes, les minutes, les heures, les jours, les mois, ou règlent l'alarme etc. L'utilisation d'une solution de multiplexage comme celle qu'on vient de décrire permet d'afficher différentes données avec seulement quelques afficheurs décimaux.

La raison d'être d'une technique de multiplexage semblable est de *partager le temps d'utilisation* des pilotes/décodeurs et des afficheurs entre deux compteurs, plutôt que d'avoir des pilotes/décodeurs et afficheurs séparés pour chaque compteur. On économise ainsi plusieurs trajets de câblage, particulièrement lorsque chaque compteur comporte un grand nombre d'étages DCB. Et ce qui est encore plus important, une telle solution diminue considérablement la consommation, parce que les pilotes/décodeurs et les afficheurs DEL prélèvent généralement un courant très intense de l'alimentation V_{CC}. Évidemment, cette

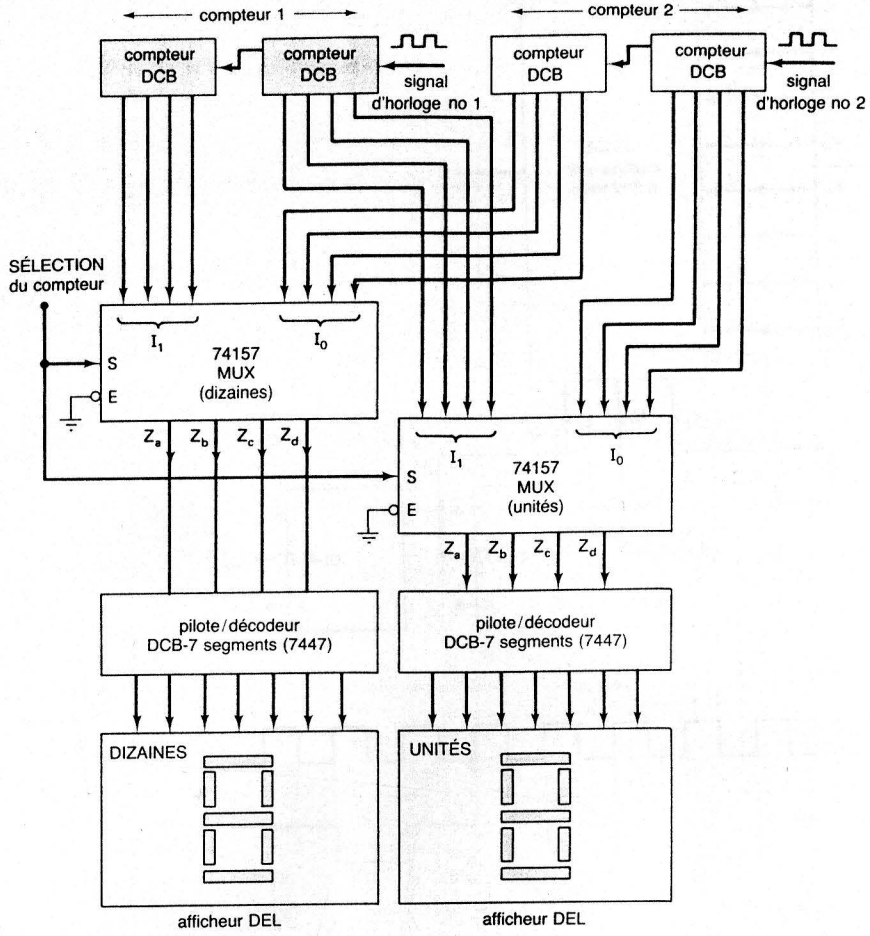


FIGURE 9-25 Système sélectif d'affichage de deux compteurs DCB à plusieurs chiffres.

technique limite l'affichage à un seul compteur à la fois. Toutefois, dans de nombreuses applications ce n'est pas un inconvénient. Il aurait été possible de construire un montage de commutation mécanique qu'on aurait actionné pour brancher un compteur ou l'autre aux pilotes/décodeurs et afficheurs; cependant, compte tenu du nombre de contacts nécessaires, de la complexité du câblage et de l'encombrement physique d'un tel montage, c'est une solution nettement plus désavantageuse que la méthode entièrement logique illustrée à la figure 9-25.

Conversion parallèle-série Dans de nombreux systèmes numériques, le traitement des données binaires se fait en parallèle (c'est-à-dire simultanément) parce que c'est une technique plus rapide. Cependant, quand on doit transférer ces données sur de longues distances, il n'est pas souhaitable de fonctionner en parallèle parce que cela exige un grand nombre

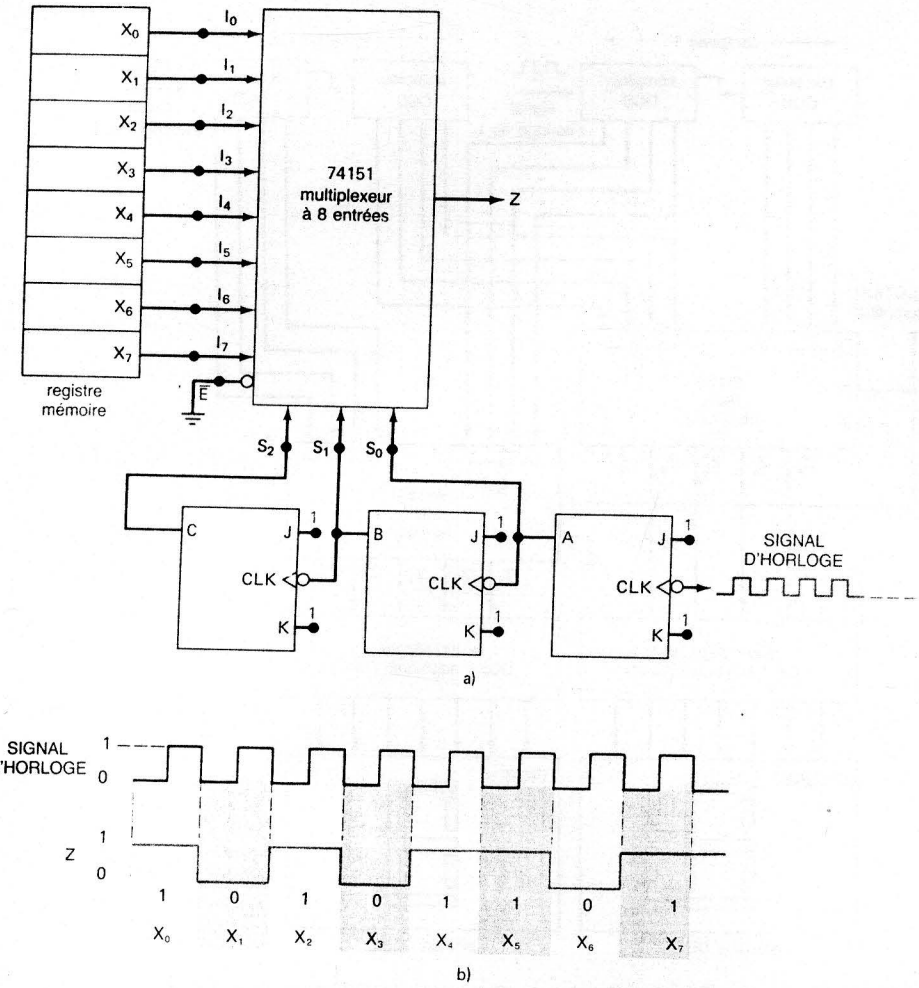


FIGURE 9-26 a) Convertisseur parallèle-série; b) formes d'ondes pour $X_7X_6X_5X_4X_3X_2X_1X_0 = 10110101$.

de lignes de transmission. C'est la raison pour laquelle les informations ou les données binaires qui sont exprimées sous forme parallèle sont souvent converties sous forme série avant d'être transmises à un endroit éloigné. Une des façons d'effectuer la *conversion parallèle-série* consiste à utiliser un multiplexeur de la manière illustrée à la figure 9-26.

Les données présentes sur les sorties du registre X sont en parallèle et viennent alimenter un multiplexeur à 8 entrées. Un compteur de 3 bits (MODULO-8) sert à produire les bits du code de sélection $S_2S_1S_0$, et ce code parcourt toutes les valeurs de 000 à 111 quand les impulsions d'horloge sont appliquées à ce compteur. De cette façon, la sortie

du multiplexeur sera X_0 pendant la première période du signal d'horloge, X_1 pendant la deuxième période, et ainsi de suite. La forme d'onde qu'on observe sur Z est une représentation série de la donnée d'entrée parallèle. La forme d'onde illustrée sur la figure correspond au cas où $X_7X_6X_5X_4X_3X_2X_1X_0 = 10110101$. Pour effectuer la conversion il faut un total de 8 cycles d'horloge. Remarquez que le bit de poids faible X_0 est transmis en premier et que le bit de poids fort X_7 est transmis en dernier.

Ordonnancement d'opérations Le circuit de la figure 9-27 est un ordonnanceur de commandes dans lequel on retrouve un multiplexeur à 8 entrées; un ordonnanceur respecte une séquence préétablie, et à chaque étape de la séquence il met en marche une certaine partie du procédé physique qui est contrôlé. Comme procédé, il peut s'agir d'un four haute température de grande dimension chauffé par sept éléments différents, allumés un à la fois. On retrouve également dans ce circuit un décodeur entrée trois voies, sortie huit voies et un compteur binaire MODULO-8. Voici la description détaillée de son fonctionnement:

1. Au départ l'état du compteur est 000. Les sorties du compteur sont délivrées aux entrées de sélection du multiplexeur et aux entrées du décodeur. Ainsi, la sortie $\overline{O}_0$ du décodeur = 0 et toutes les autres sont à 1, de sorte que les entrées de l'ACTIONNEUR du procédé sont toutes BASSES. Les sorties des CAPTEURS du procédé sont donc toutes, au départ, au niveau BAS. La sortie du multiplexeur $\overline{Z} = \overline{I}_0 = 1$, puisque les entrées S sont 000.
2. L'impulsion DÉBUT lance l'opération d'ordonnancement en mettant la bascule Q_0 au niveau HAUT, ce qui installe dans le compteur l'état 001. La sortie $\overline{O}_1$ du décodeur se trouve maintenant au niveau BAS, ce qui déclenche l'actionneur 1 correspondant à la première étape du procédé (il s'agit peut-être d'allumer un élément chauffant).
3. Un peu plus tard, la sortie de CAPTEUR 1 passe au niveau HAUT, ce qui indique la fin de la première étape (correspondant peut-être à l'atteinte d'un certain niveau de température). Ce niveau HAUT se retrouve à présent à l'entrée I_1 du multiplexeur. Il est inversé et se retrouve sur la sortie $\overline{Z}$ puisque le code de sélection du compteur est 001.
4. La valeur BASSE sur $\overline{Z}$ est appliquée à l'entrée *CLK* de la bascule Q_0. Ce front descendant incrémente le compteur jusqu'à l'état 010.
5. La sortie du décodeur $\overline{O}_2$ passe maintenant au niveau BAS, ce qui déclenche l'actionneur 2 correspondant à la seconde étape du procédé. $\overline{Z}$ égale maintenant $\overline{I}_2$ (code de sélection est 010). Puisque la sortie de CAPTEUR 2 est encore à 0, $\overline{Z}$ passe au niveau 1.
6. Quand la deuxième étape du procédé est achevée, la sortie de CAPTEUR 2 passe au niveau HAUT, produisant ainsi un niveau BAS sur $\overline{Z}$ et faisant incrémenter le compteur jusqu'à 011.
7. La même chose se répète pour chacune des autres étapes. À la fin de la septième étape, la sortie de CAPTEUR 7 passe au niveau HAUT, amenant ainsi le compteur de 111 à 000, état où il demeure jusqu'à ce qu'une autre impulsion DÉBUT réamorce la séquence.

Génération d'une fonction logique Il est possible d'utiliser des multiplexeurs pour matérialiser directement des fonctions logiques à partir d'une table de vérité sans devoir passer par le processus de simplification. Quand on utilise un multiplexeur à cette fin, les entrées de sélection reçoivent les variables logiques et chaque entrée de données est raccordée en permanence à un 0 ou à un 1 de façon à respecter la table de vérité.

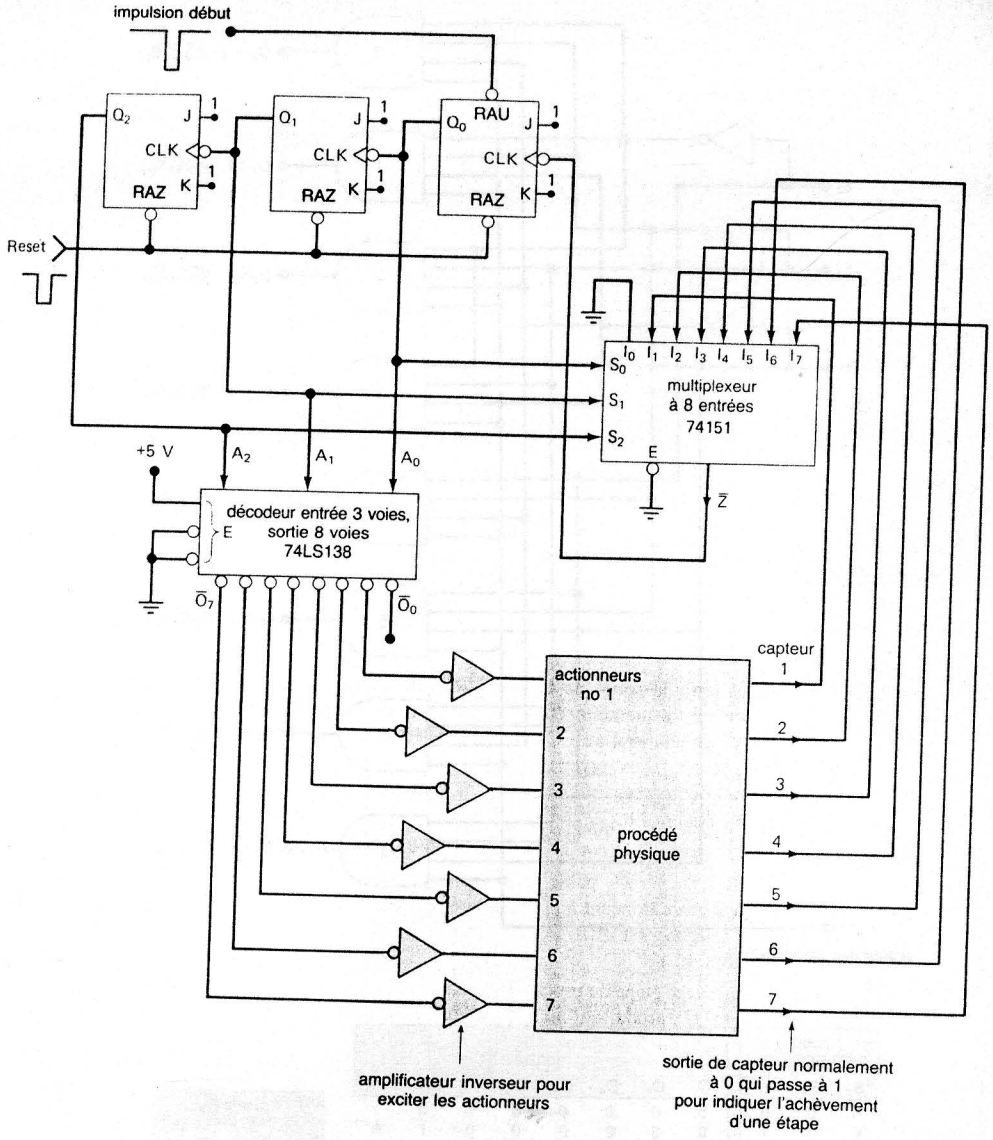
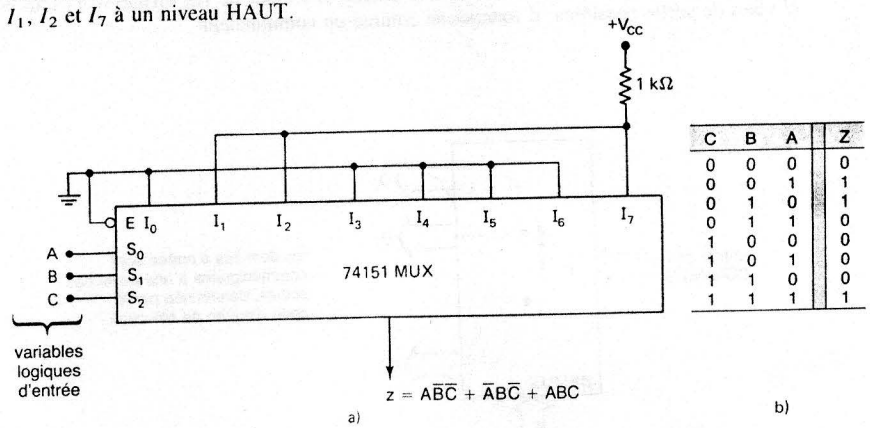


FIGURE 9-27 Ordonnanceur de commandes à sept étapes.

La figure 9-28 nous montre comment on utilise un multiplexeur à 8 entrées pour matérialiser un circuit logique qui fonctionne conformément aux indications de la table de vérité. Les variables d'entrée A, B, C sont raccordées respectivement à S_0, S_1, S_2, et ce sont les niveaux appliqués à ces entrées qui déterminent l'entrée de données qui est transférée à la sortie Z. Conformément à cette table de vérité, Z doit être à 0 quand $CBA = 000$. Ainsi l'entrée du multiplexeur I_0 doit être raccordée à 0. De la même façon, Z est supposée être à 0 quand $CBA = 011$, 100, 101 et 110, de sorte que les entrées I_3, I_4, I_5 et I_6 sont raccordées en permanence à la masse (niveau BAS). L'autre groupe de conditions CBA doit donner $Z = 1$, de sorte qu'il faut raccorder en permanence les entrées du multiplexeur I_1, I_2 et I_7 à un niveau HAUT.



C	B	A	Z
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

FIGURE 9-28 Utilisation d'un multiplexeur pour matérialiser la fonction logique décrite par la table de vérité.

Il est facile de voir que toute table de vérité à 3 variables peut être matérialisée au moyen d'un multiplexeur à 8 entrées. Cette méthode de matérialisation est souvent beaucoup plus efficace que l'utilisation de portes logiques distinctes. Par exemple, si nous écrivons la somme de produits correspondant à la table de vérité de la figure 9-28, nous trouvons

$$Z = A\overline{B}\overline{C} + \overline{A}B\overline{C} + ABC$$

Il est *impossible* de simplifier cette expression au moyen de l'algèbre ou d'un diagramme-K, ce qui signifie que pour la matérialiser il faut 3 INVERSEURS et 4 portes NON-ET, soit au total 2 circuits intégrés.

Il existe une méthode encore plus efficace, soit réaliser de telles fonctions logiques en se servant de multiplexeurs. Cette méthode permet à un concepteur logicien de prendre un multiplexeur à trois entrées de sélection (par ex. le 74151) et de construire une fonction logique à *quatre variables*. Nous présentons les détails de cette méthode au problème 9-35.

QUESTIONS DE RÉVISION

1. Dites quelles sont les principales applications des multiplexeurs.
2. *Vrai ou faux:* Quand un multiplexeur est utilisé pour matérialiser une fonction logique, les variables booléennes sont appliquées aux bornes de données du multiplexeur.
3. Quel type de circuit délivre les entrées de sélection à un multiplexeur quand on s'en sert comme convertisseur parallèle-série?

9-9 DÉMULTIPLEXEURS (DISTRIBUTEURS DE DONNÉES)

On vient de voir qu'un multiplexeur est sollicité par plusieurs entrées, mais ne transmet *qu'une* de ces dernières à la sortie. Un *démultiplexeur* effectue l'opération inverse: il n'a qu'une entrée et dirige celle-ci vers une sortie parmi plusieurs sorties. La figure 9-29 nous montre le schéma général d'un démultiplexeur (DEMUX). Les flèches plus larges utilisées pour représenter les entrées et les sorties signifient plusieurs conducteurs. Le code d'entrée de sélection détermine à quelle sortie l'entrée DONNÉES est transmise. Autrement dit, le démultiplexeur reçoit des données d'une entrée et choisit de les diriger vers une des N voies de sortie possibles; il fonctionne comme un commutateur.

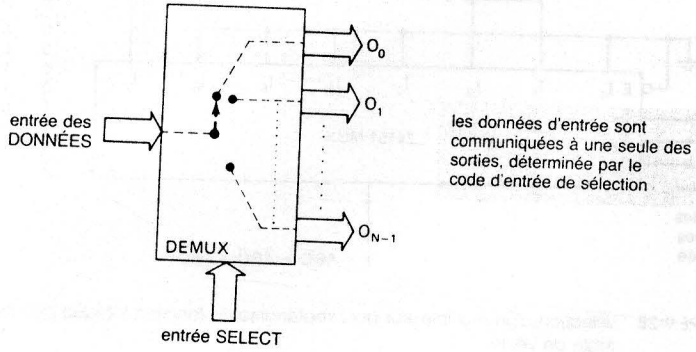
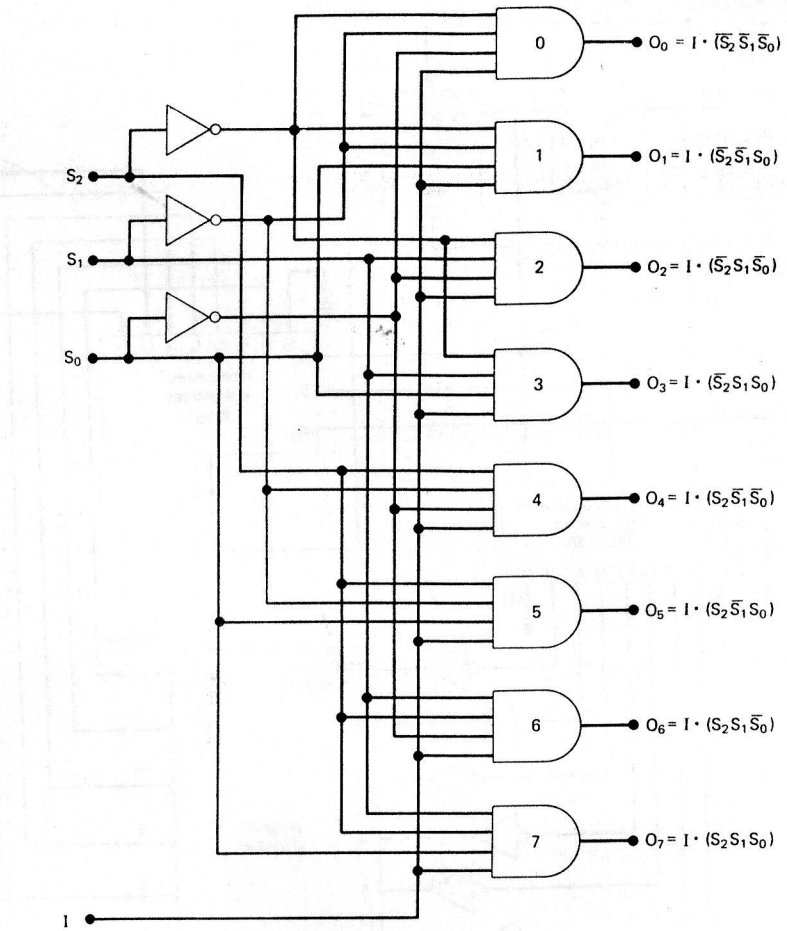


FIGURE 9-29 Schéma général d'un démultiplexeur.

Démultiplexeur entrée une voie, sortie huit voies La figure 9-30 nous fait voir le schéma logique d'un démultiplexeur qui dirige une voie d'entrée vers huit voies de sortie possibles. La ligne d'entrée de données unique I est connectée aux huit portes ET, mais une seule de ces portes est validée par les entrées SELECT. Par exemple, quand $S_2S_1S_0 = 000$, seule la porte ET 0 est validée, et l'entrée de données I se retrouve sur la sortie O_0. Les autres codes SELECT font en sorte que l'entrée I traverse les portes des autres sorties. La table de vérité récapitule ce fonctionnement.

Le circuit du démultiplexeur de la figure 9-30 est très semblable à celui du décodeur entrée trois voies, sortie huit voies de la figure 9-2, si on excepte le fait qu'une quatrième entrée (I) a été ajoutée à chaque porte. Nous avons déjà souligné que de nombreux décodeurs intégrés ont une entrée VALIDATION, et que ceci correspond à une entrée supplémentaire pour les portes du décodeur. Ce genre de boîtier de décodeur peut donc servir de démultiplexeur, les entrées du code binaire (par exemple A, B, C de la figure 9-2) jouant le rôle des entrées SELECT et l'entrée VALIDATION jouant le rôle de l'entrée de données I. C'est donc une des raisons pour laquelle les fabricants de CI donnent souvent le nom de **décodeur/démultiplexeur** à ce dispositif puisqu'il peut servir à l'une ou l'autre de ces fonctions.



code SÉLECTION			sorties							
S_2	S_1	S_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
0	0	0	0	0	0	0	0	0	0	I
0	0	1	0	0	0	0	0	0	I	0
0	1	0	0	0	0	0	0	I	0	0
0	1	1	0	0	0	0	I	0	0	0
1	0	0	0	0	0	I	0	0	0	0
1	0	1	0	0	I	0	0	0	0	0
1	1	0	0	I	0	0	0	0	0	0
1	1	1	I	0	0	0	0	0	0	0

Note: I est l'entrée de données

FIGURE 9-30 Démultiplexeur entrée une voie, sortie huit voies.

Nous avons déjà étudié le 74LS138 dans son rôle de décodeur un parmi huit. La figure 9-31 nous montre maintenant comment l'utiliser à titre de démultiplexeur. L'entrée de validation $\overline{E}_1$ est utilisée comme l'entrée de données I, tandis que les deux autres entrées de validation sont gardées en permanence à leur niveau vrai. Les entrées $A_2A_1A_0$ accueillent le code de sélection. Pour illustrer son fonctionnement, supposons que les entrées de sélection sont 000. On sait que ce code ouvre seulement la sortie appelée $\overline{O}_0$ et garde toutes les autres sorties au niveau HAUT. $\overline{O}_0$ passe à 0 seulement si $\overline{E}_1$ passe à 0, et cette sortie passe à 1 si $\overline{E}_1$ passe à 1. Autrement dit, $\overline{O}_0$ suit les niveaux appliqués à la borne $\overline{E}_1$ (c'est-à-dire l'entrée de données I). Pendant ce temps toutes les autres sorties demeurent à 1. De la même façon, un code de sélection différent appliqué aux entrées $A_2A_1A_0$ ouvre la sortie correspondante qui alors suit les valeurs appliquées sur l'entrée de données I.

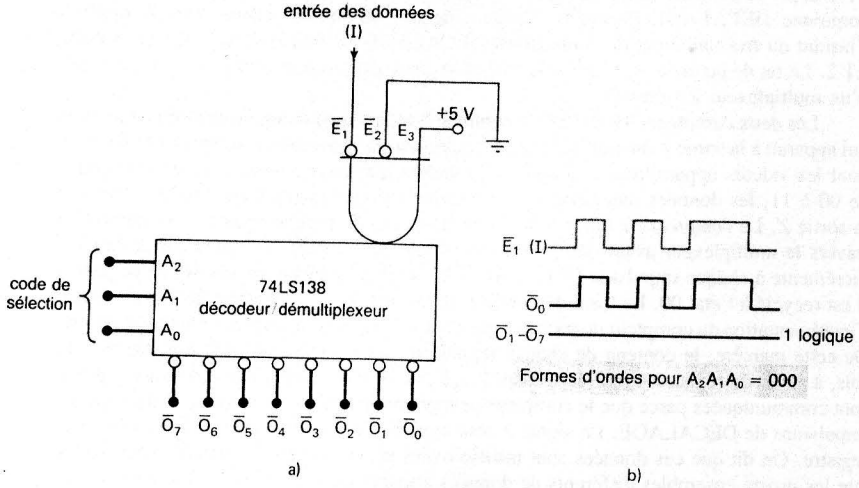


FIGURE 9-31 a) Le décodeur 74LS138 peut agir comme multiplexeur si on détermine que $\overline{E}_1$ sera l'entrée de données; b) formes d'ondes types quand le code de sélection est $A_2A_1A_0$ = 000, la sortie $\overline{O}_0$ est identique à l'entrée de données.

La figure 9-31 b) montre les formes d'ondes types pour le cas sélectionné par $A_2A_1A_0 = 000$, soit la sortie $\overline{O}_0$. Dans cet exemple, le signal de données appliqué sur $\overline{E}_1$ est acheminé sur $\overline{O}_0$ et toutes les autres sorties restent dans leur état inactif, soit l'état HAUT.

Démultiplexeur d'horloge Le principe du démultiplexage trouve de nombreuses applications un peu partout. La figure 9-32 nous fait voir le démultiplexeur 74LS138 utilisé à titre de ***démultiplexeur d'horloge***. Commandé par les lignes SELECT, ce dispositif aiguille le signal d'horloge vers la destination voulue. Par exemple, quand $S_2S_1S_0 = 000$, le signal d'horloge appliqué à I est passé à la sortie $\overline{O}_0$. $S_2S_1S_0 = 101$, l'horloge est passée à la sortie $\overline{O}_5$.

Système de surveillance de la sécurité des lieux Examinons la situation d'un système de surveillance de la sécurité des lieux dans une usine industrielle où l'état ouvert/fermé

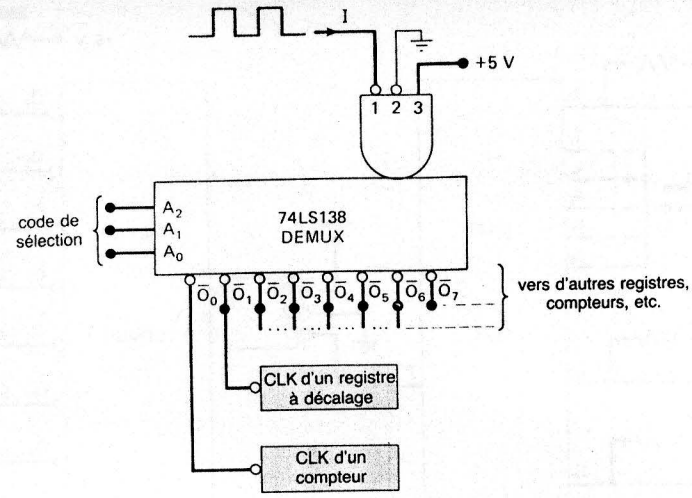


FIGURE 9-32 Un démultiplexeur d'horloge achemine le signal d'horloge à la sortie désignée par les bornes du code de sélection.

d'un grand nombre de portes d'accès est continuellement sous surveillance. Chaque porte détermine l'état d'un interrupteur; chaque interrupteur doit être associé à une DEL dans la console de surveillance du gardien de sécurité. Une façon de parvenir à un tel montage est de faire courir un conducteur de chaque interrupteur vers son voyant témoin dans la console de télésurveillance. Cela exigera l'installation de nombreux conducteurs sur de longues distances. Une solution qui allège le câblage dans la console est le recours à un couple multiplexeur/démultiplexeur. La figure 9-33 nous fait voir un système de télésurveillance de huit portes; évidemment, le principe exposé peut être appliqué à un nombre quelconque de portes.

EXEMPLE 9-11

Examinez attentivement la figure 9-33 et décrivez le fonctionnement complet de ce système.

Solution Les huit interrupteurs de porte constituent les entrées de données du MUX; ils produisent un niveau HAUT quand la porte est ouverte et un niveau BAS, quand elle est fermée. Le compteur MOD-8 fournit les entrées de sélection au MUX et au DEMUX de la console de télésurveillance. Chaque sortie du DEMUX est raccordée à un voyant DEL qui est allumé quand la sortie est BASSE. Les impulsions d'horloge appliquées au compteur font parcourir à ce dernier la suite des états possibles de 000 à 111. Pour chaque nombre du compteur, l'état de l'interrupteur de la porte de même numéro est inversé par le MUX qui le passe à sa sortie $\overline{Z}$. Depuis cette sortie, l'état est transmise à l'entrée du DEMUX qui le communique à la sortie ayant le numéro de cette porte.

Par exemple, supposons que le compteur contienne le nombre 110 (6). Pendant que le compteur est dans cet état, la porte 6 est fermée. Le niveau BAS sur I_6 traverse le MUX et est inversé, de sorte que l'on retrouve un niveau HAUT sur $\overline{Z}$. Ce niveau HAUT est

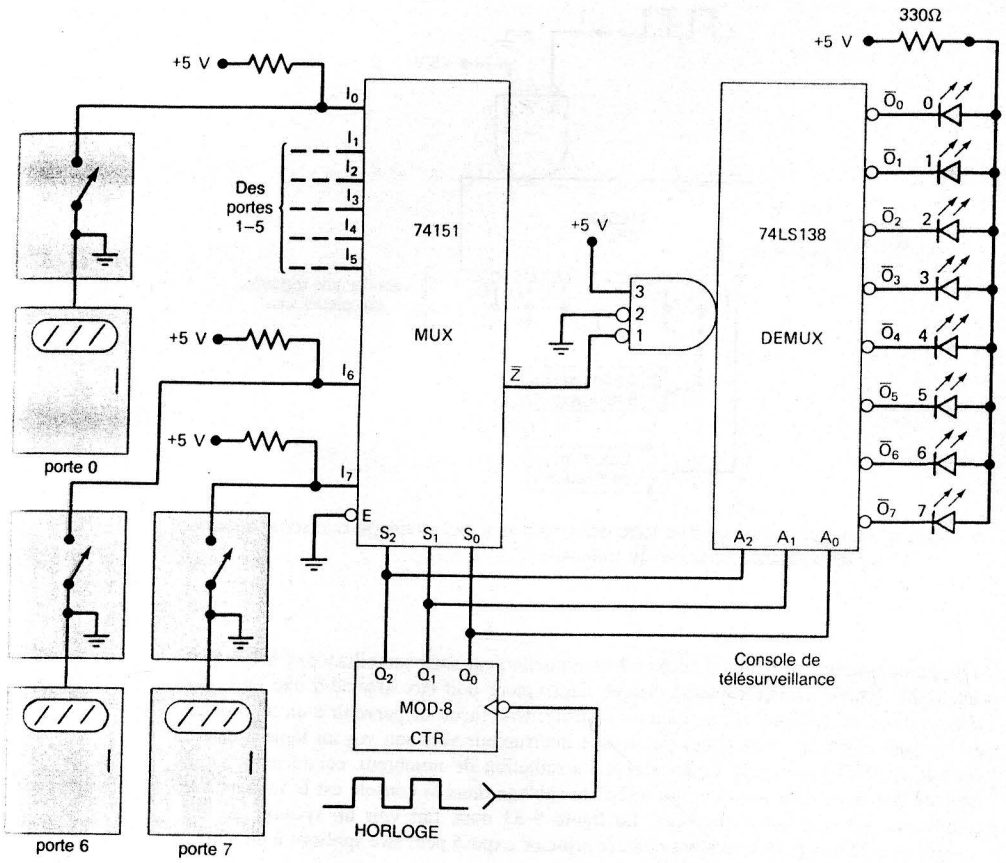


FIGURE 9-33 Système de surveillance de la sécurité des lieux.

est passé, au travers du DEMUX, jusqu'à la sortie $\overline{O}_6$, ce qui a pour effet d'éteindre le voyant DEL 6 et d'indiquer que la porte 6 est fermée. Mainenant, supposons que la porte 6 est ouverte. C'est à présent un niveau BAS qui se trouve sur $\overline{Z}$ et sur O_6, ce qui allume la DEL 6 afin de signaler que la porte 6 est ouverte. Évidemment, tous les autres voyants DEL sont éteints, puisqu'à cet instant la seule sortie active est $\overline{O}_6$.

Pendant que le compteur est cadencé par l'horloge et parcourt les huit états de 000 à 111, les DEL indiquent séquentiellement l'état des huit portes. Si toutes les portes sont fermées, aucune des DEL ne s'allumera au moment de la sélection de leur sortie du DEMUX. Si une porte est ouverte, son voyant DEL s'allume durant l'intervalle de temps où le compteur contient son numéro; il est éteint à tous les autres moments. Par conséquent, une porte ouverte est signalée par un voyant qui clignote. La vitesse de clignotement peut être réglée en agissant sur la fréquence de l'horloge.

Notez qu'il n'y a que quatre conducteurs de signaux qui quittent les circuits de « capteur de porte » et aboutissent à la console de télésurveillance: la sortie $\overline{Z}$ et les trois lignes de sélection. Il s'agit donc d'une économie de quatre conducteurs par rapport à l'autre solution d'un conducteur par porte. Le couple multiplexeur/démultiplexeur transmet l'état de chaque porte à son voyant un à la fois (en série) plutôt que tout d'un bloc (en parallèle).

Système de transmission des données synchrone Le schéma logique reproduit à la figure 9-34 est celui d'un système de transmission de données synchrone grâce auquel on transmet en série des mots de 4 bits entre un émetteur et un récepteur à distance. Examinons d'abord les circuits de l'émetteur. Les mots binaires sont mémorisés dans les registres *A*, *B*, *C* et *D*, eux-mêmes câblés comme des registres à décalage circulaire ayant une entrée commune DÉCALAGE (horloge). Chaque registre décale ses valeurs vers la droite à l'instant du front montant des impulsions DÉCALAGE provenant de la sortie de la porte ET 2. Le bit de poids le plus faible de chaque registre est raccordé à une entrée de données d'un multiplexeur à 4 entrées.

Les deux compteurs MODULO-4 commandent la transmission du contenu du registre qui apparaît à la sortie *Z* du multiplexeur. Le *compteur de mots* choisit le registre de données dont les valeurs apparaîtront à la sortie Z. Pendant que ce compteur parcourt les 4 états de 00 à 11, les données mémorisées dans chaque registre apparaissent successivement à la sortie Z. Le *compteur de bits* s'assure que les 4 bits de chaque registre sont transmis à travers le multiplexeur avant que le registre suivant soit appelé. Le compteur de bits est incrémenté à chaque impulsion DÉCALAGE, de sorte qu'après 4 impulsions de ce genre, il est recyclé à l'état 00. Le front descendant de la sortie Q_1 du compteur de bits provoque l'incrémentation du compteur de mots, ce qui choisit le registre suivant pour la transmission. De cette manière, le contenu de chaque registre de données est transmis à *Z*, un bit à la fois, à partir du registre *A* (correspondant à $S_1S_0 = 00$); et les données de chaque registre sont communiquées parce que le compteur de mots progresse d'une valeur à chaque quatre impulsions de DÉCALAGE. Le signal *Z* sera donc une donnée série de 16 bits, 4 bits par registre. On dit que ces données sont **multiplexées par répartition dans le temps** parce que les quatre ensembles différents de données apparaissent sur la même ligne de sortie, mais à des moments différents.

Le processus de transmission est commandé par deux bascules *D*, les portes ET 1 et 2 et le monostable. Un des paragraphes suivants est consacré à la description du fonctionnement de la logique de ce circuit.

Le récepteur Les circuits du récepteur contiennent un démultiplexeur un parmi quatre qui reçoit le signal *Z* envoyé par le multiplexeur de l'émetteur et le *démultiplexe*; c'est-à-dire qu'il sépare les quatre ensembles différents de données et les dirige vers quatre sorties différentes de façon à ce que la donnée qui provient du registre *A* apparaisse un bit après l'autre sur O_0, que la donnée provenant du registre *B* apparaisse sur O_1, et ainsi de suite. Le résultat définitif est presque le même que celui d'avoir chaque registre de données de l'émetteur connecté à sa sortie correspondante du récepteur, la seule différence étant que les données sont envoyées un registre à la fois par l'entremise d'une voie de transmission de données série.

Les compteurs MODULO-4 du récepteur ont la même fonction que leur symétrique dans l'émetteur. Le compteur de mots choisit la sortie du démultiplexeur qui reçoit la donnée, et le compteur de bits permet à 4 bits de données d'atteindre une sortie avant de faire passer le compteur de mots à son état suivant. Les fonctions de la bascule, du monostable et de la porte ET sont décrites ci-après.

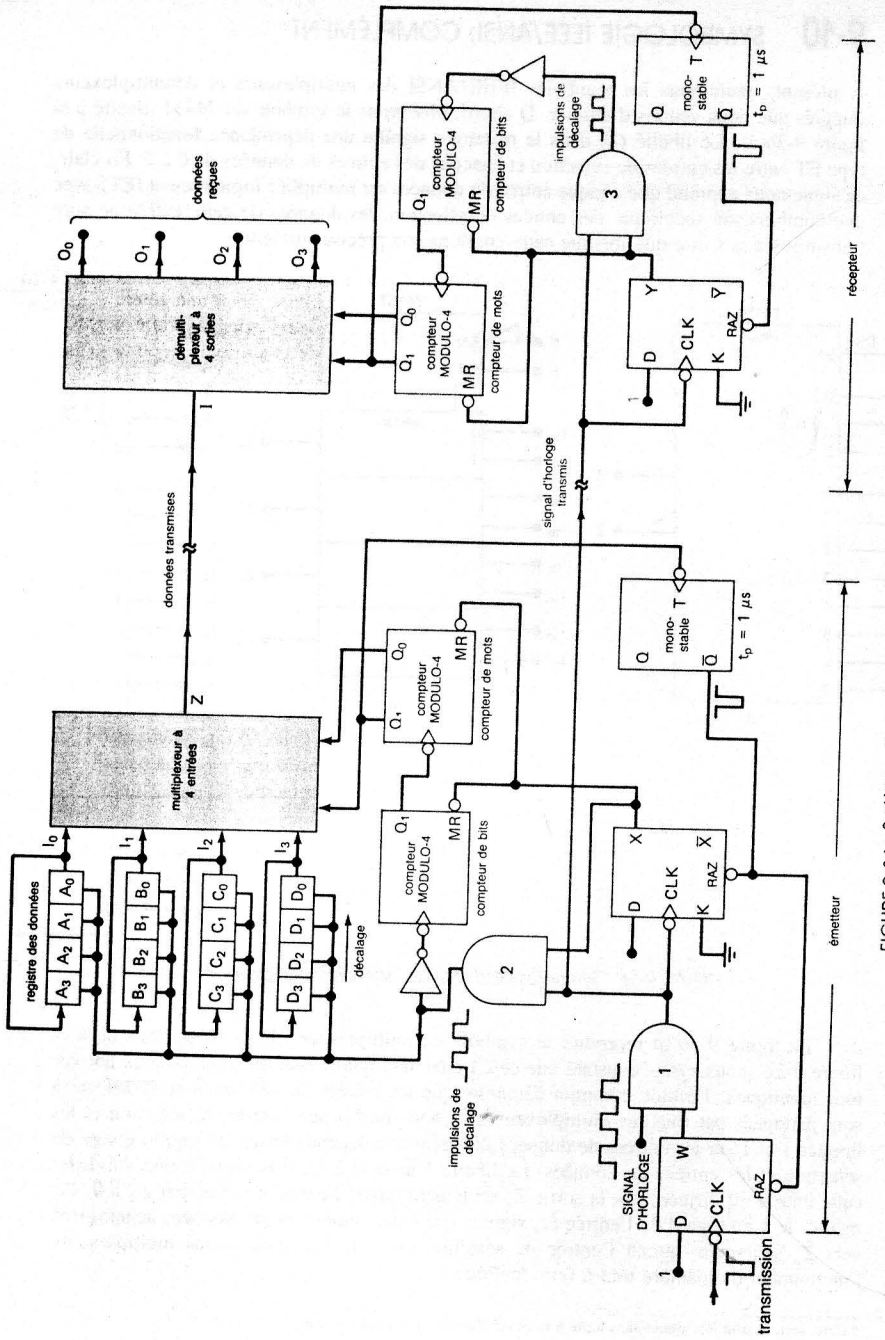


FIGURE 9-34 Système de transmission de données synchrone.

Fonctionnement complet Il est bien évident que pour qu'un tel système de transmission de données fonctionne comme il se doit, on doit prévoir une façon de synchroniser le choix des entrées du multiplexeur de l'émetteur avec le choix des sorties du démultiplexeur du récepteur. Nous allons parcourir tout un cycle de fonctionnement afin de voir comment cette synchronisation s'accomplit. Pour les besoins de cet exemple, nous supposerons que les données qui se trouvent dans les registres sont:

$$[A] = 0110, \quad [B] = 1001, \quad [C] = 1011, \quad [D] = 0100$$

À mesure que nous franchirons les différentes étapes, nous nous référerons aux formes d'ondes de la figure 9-35.

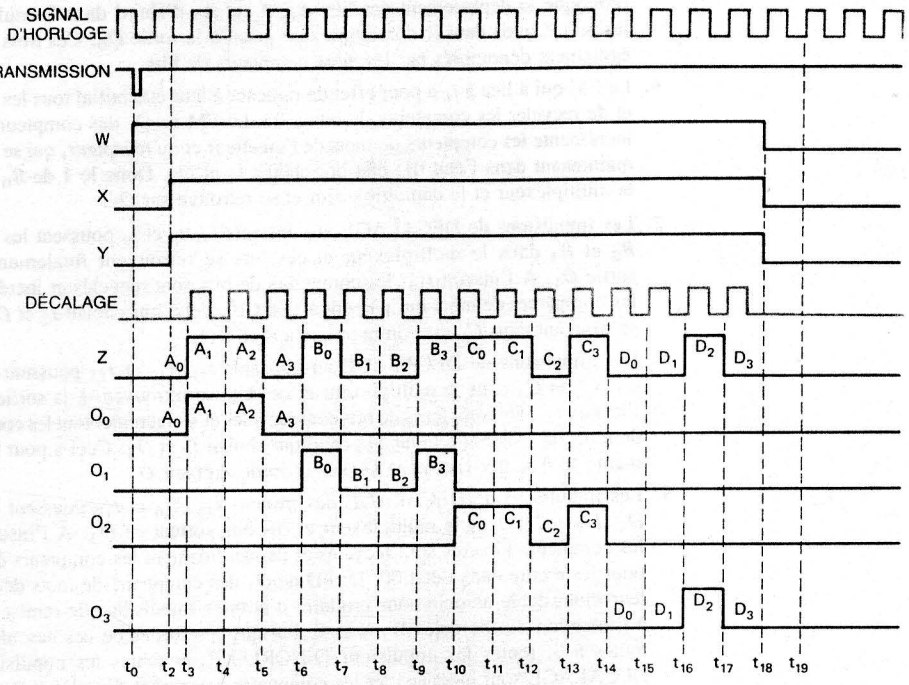


FIGURE 9-35 Formes d'ondes qui se produisent durant un cycle complet de la transmission.

1. Les sorties des bascules W et X de l'émetteur et de la bascule Y du récepteur sont normalement au niveau BAS. Ces 0 sur X et Y gardent les deux ensembles de compteurs dans leur état 0. Le 0 sur W empêche les impulsions du SIGNAL D'HORLOGE de traverser la porte ET 1.
2. Comme les deux compteurs de mots sont à 00, le 0 en A_0 traverse le multiplexeur pour arriver jusqu'à Z et de là est communiqué à travers le démultiplexeur à la sortie O_0. Toutes les autres sorties du démultiplexeur sont à 0, étant donné qu'elles n'ont pas été choisies.
3. Voilà la situation avant t_0. À l'instant t_0, l'impulsion TRANSMISSION met $W = 1$ ce qui valide la porte ET 1 et laisse passer les impulsions du SIGNAL D'HORLOGE. Ces impulsions sont également celles du SIGNAL D'HOR-

LOGE transmis envoyé au récepteur en même temps que les DONNÉES TRANSMISES.

4. Le front descendant de la première impulsion qui sort de la porte ET 1 installe un niveau HAUT dans les bascules X et Y à l'instant t_2. Ceci débloque les compteurs, qui étaient gardés à 0, et valide les portes ET 2 et 3 qui laissent passer les impulsions du SIGNAL D'HORLOGE à partir de l'instant t_3. Les impulsions qui franchissent la porte ET 2 sont les impulsions de DÉCALAGE. Les impulsions qui franchissent la porte ET 3 sont identiques aux impulsions de DÉCALAGE.
5. Le front montant des impulsions de DÉCALAGE aux instants t_3, t_4 et t_5 provoque le déplacement des bits A_1, A_2 et A_3 d'abord dans le multiplexeur jusqu'à Z, puis dans le démultiplexeur jusqu'à la sortie O_0. Ces trois FM sont également dénombrés par les deux compteurs de bits.
6. Le FM qui a lieu à t_6 a pour effet de ramener à leur état initial tous les registres et de recycler les compteurs de bits à 00. Le FM de Q_1 des compteurs de bits incrémente les compteurs de mots de l'émetteur et du récepteur, qui se trouvent maintenant dans l'état 01, état qui choisit I_1 et O_1. Donc le 1 de B_0 traverse le multiplexeur et le démultiplexeur et se retrouve sur O_1.
7. Les impulsions de DÉCALAGE aux instants t_7, t_8 et t_9 poussent les bits B_1, B_2 et B_3 dans le multiplexeur et ces bits se retrouvent finalement sur la sortie O_1. À l'instant t_{10}, les compteurs de bits sont recyclés et incrémentent les compteurs de mots qui passent à l'état 10, celui qui choisit I_2 et O_2. Le 1 se trouvant dans C_0 est communiqué à Z et à O_2.
8. Les impulsions de DÉCALAGE aux instants t_{11}, t_{12} et t_{13} poussent les bits C_1, C_2 et C_3 dans le multiplexeur et ces bits se retrouvent à la sortie O_2. À l'instant t_{14}, les compteurs de bits sont recyclés et ils incrémentent les compteurs de mots qui passent à l'état 11, celui qui choisit I_3 et O_3. Ceci a pour effet de mettre le 0 se trouvant dans D_0 sur la sortie Z et sur O_3.
9. Les impulsions de DÉCALAGE des instants t_{15}, t_{16} et t_{17} poussent les bits D_1, D_2 et D_3 dans le multiplexeur et ces bits sortent en O_3. À l'instant t_{18}, les compteurs binaires sont recyclés et ils incrémentent les compteurs de mots pour les mettre dans l'état 00. Le FD de Q_1 des compteurs de mots déclenche leur monostable associé pour produire d'étroites impulsions de remise à zéro à l'intention des bascules W, X et Y. Puisque les sorties de ces bascules sont toutes à 0, toutes les impulsions D'HORLOGE et toutes les impulsions de DÉCALAGE sont bloquées, et les compteurs demeurent dans l'état 0.
10. Les conditions du circuit sont revenues à leur état original. Aucune donnée ne sera transmise avant l'arrivée de la prochaine impulsion TRANSMISSION.

Les formes d'ondes observées sur Z et sur O_0–O_3 démontrent comment les données des registres sont multiplexées pour former le signal Z et ensuite démultiplexées de façon à ce que chacune des sorties reçoive la donnée exacte.

QUESTIONS DE RÉVISION

1. Expliquez la différence entre un DÉMULTIPLEXEUR et un MULTIPLEXEUR.
2. *Vrai ou faux:* Le circuit de base d'un DÉMULTIPLEXEUR est essentiellement le même que celui d'un MULTIPLEXEUR.
3. Pour le système de la figure 9-33, que voit le gardien de sécurité à sa console si toutes les portes sont ouvertes?

9-10 SYMBOLOGIE IEEE/ANSI: COMPLÉMENT*

À présent, examinons les symboles IEEE/ANSI des multiplexeurs et démultiplexeurs intégrés que nous venons d'étudier. D'abord, observons le symbole du 74151 illustré à la figure 9-36 a). Le libellé $G\frac{0}{7}$ dans le rectangle signifie une dépendance fonctionnelle de type ET entre les entrées de sélection et chacune des entrées de données de 0 à 7. En clair, ce signe nous apprend que chaque entrée de données est multipliée logiquement (ET) avec une combinaison spécifique des entrées de sélection; les données de cette entrée ne sont transmises à la sortie que lorsque cette combinaison précise survient.

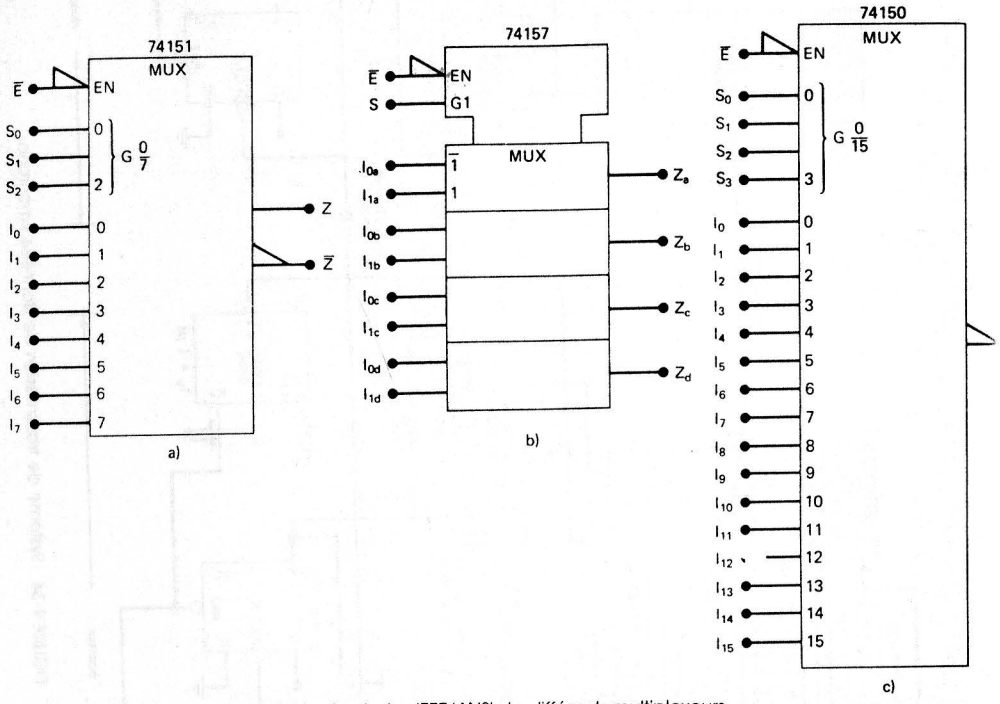


FIGURE 9-36 Symboles IEEE/ANSI de différents multiplexeurs.

La figure 9-36 b) reproduit le symbole du multiplexeur 74157. Rappelons qu'à la figure 9-24, nous avons constaté que ce CI contenait quatre multiplexeurs à deux entrées tous identiques. Le bloc commun démontre que les entrées de validation et de sélection sont partagées par tous les multiplexeurs. La notation G1 sur l'entrée de sélection et les libellés $\overline{1}$ et 1 sur les entrées de données illustrent une dépendance en ET entre l'entrée de sélection et les entrées de données. Le libellé $\overline{1}$ associé à I_{0a} a la signification suivante: cette entrée est dirigée vers la sortie Z_a seulement quand l'entrée de sélection est à 0. De même, le 1 en regard de l'entrée I_{1a} signale que cette entrée voit ses données acheminées vers Z_a seulement quand l'entrée de sélection est à 1. Les trois autres multiplexeurs fonctionnent de manière tout à fait identique.

* Cette section peut être omise sans nuire à la compréhension des parties à venir.

EXEMPLE 9-12

Le symbole IEEE/ANSI du CI 74150 est montré à la figure 9-36 c). Quelle est la fonction de ce dispositif?

Solution Le 74150 est un multiplexeur à 16 entrées. Remarquez les quatre entrées de sélection qui établissent laquelle parmi les 16 entrées voit ses données transmises à la sortie. Prenez note également que la sortie est inversée, de sorte que les données de sortie sont l'inverse des données d'entrée.

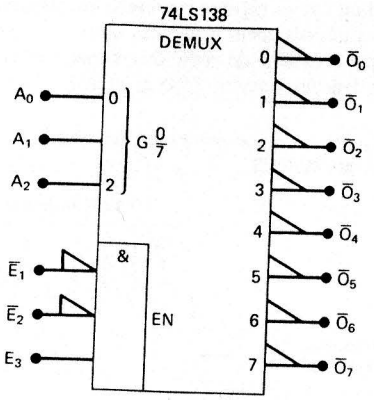


FIGURE 9-37 Le symbole IEEE/ANSI du démultiplexeur 74LS138.

9-11 SUPPLÉMENT DE DÉPANNAGE

Nous poursuivons notre illustration de la technique observation et raisonnement, phase primordiale au début de toute action de dépannage, en analysant trois exemples supplémentaires. Pour chaque exemple, nous vous demandons de rechercher la défaillance du circuit avant de consulter la solution.

EXEMPLE 9-13

Examinez le circuit de la figure 9-25. Un test effectué sur ce circuit a produit les résultats suivants:

		Nombre réel	Nombre affiché
Cas 1	Compteur 1	25	25
	Compteur 2	37	35
Cas 2	Compteur 1	49	49
	Compteur 2	72	79
Cas 3	Compteur 1	96	96
	Compteur 2	14	16

Quelle est la cause de ce dysfonctionnement?

Solution Dans chacun des cas du test, l'affichage du compteur 1 concorde avec le nombre réel. Il faut en conclure que les entrées I_1, que toutes les sorties des multiplexeurs, et que les deux afficheurs fonctionnent correctement. En revanche, pour chaque cas, le chiffre des *dizaines* du compteur 2 est exact, tandis que le chiffre des *unités* est fautif. La défaillance devrait donc se situer quelque part entre la sortie de la section des unités du compteur 2 et les entrées I_0 des multiplexeurs. L'action suivante serait donc de comparer le code binaire des valeurs réelles et affichées qui proviennent des unités du compteur 2.

	Unités réelles	Unités affichées
Cas 1	0111 (7)	0101 (5)
Cas 2	0010 (2)	1001 (9)
Cas 3	0100 (4)	0110 (6)

Le principe derrière tout cela est de trouver un bit qui ne varie pas (reste continuellement à 1 ou à 0), ou bien deux bits intervertis (connexions croisées). Les données ci-dessus ne révèlent rien de ce genre.

Si on analyse plus en profondeur ces mesures, on s'aperçoit que le chiffre des unités du compteur 2 affiché est toujours identique au chiffre de l'unité du compteur 1. Ce symptôme résulte probablement d'un HAUT constant sur l'entrée de sélection du multiplexeur des unités, car cela a pour conséquence de toujours passer le chiffre des unités du compteur 1 à la sortie du multiplexeur des unités. Ce HAUT constant sur l'entrée de sélection a sans doute pour cause un conducteur ouvert entre l'entrée de sélection du multiplexeur des dizaines et l'entrée de sélection du multiplexeur des unités. Il est impossible que ce niveau HAUT provienne d'un court-circuit à V_{CC}, puisque dans un tel cas, l'entrée de sélection du multiplexeur des dizaines serait en permanence à 1; ce qui n'est pas le cas, car nous savons que le multiplexeur des dizaines fonctionne.

EXEMPLE 9-14

Le système de surveillance de la sécurité des lieux de la figure 9-33 est testé et les résultats sont reportés dans le tableau que voici:

Condition	DEL
Toutes les portes fermées	Toutes éteintes
Porte 0 ouverte	DEL 4 clignote
Porte 1 ouverte	DEL 5 clignote
Porte 2 ouverte	DEL 6 clignote
Porte 3 ouverte	DEL 7 clignote
Porte 4 ouverte	DEL 4 clignote
Porte 5 ouverte	DEL 5 clignote
Porte 6 ouverte	DEL 6 clignote
Porte 7 ouverte	DEL 7 clignote

Quelle est la cause possible d'un tel comportement?

Solution De nouveau, il convient de bien examiner les résultats afin de repérer un certain motif qui pourrait nous aider à circonscrire la recherche à une petite zone du circuit. Les données de ci-dessus nous apprennent que les bons voyants clignotent quand les portes 4

à 7 sont ouvertes. En plus, on remarque que pour les portes 0 à 3, le numéro du voyant qui clignote est de *quatre* au-delà du numéro de la porte; les voyants 0 à 3 ne clignotent jamais. Tout cela est probablement causé par un niveau logique HAUT constant sur A_2, le bit de poids le plus fort des entrées de sélection du démultiplexeur, parce que quand c'est le cas, seuls les codes 4 et supérieurs sont appliqués; en outre, ce HAUT constant additionne un 4 aux codes de sélection de 0 à 3.

Deux possibilités s'offrent donc: ou bien A_2 est court-circuité à V_{CC}, ou bien il y a une connexion ouverte dans A_2. Un peu de réflexion nous amène toutefois à éliminer la première possibilité, car si cela était, S_2 du multiplexeur serait toujours HAUT, ce qui aurait pour effet d'empêcher le signal d'état des portes 0 à 3 d'arriver jusqu'au multiplexeur, de le franchir et de parvenir dans le démultiplexeur. Nous savons que ce n'est pas le cas, puisque les observations relevées montrent que lorsqu'une de ces portes est ouverte, il se passe quelque chose sur une des sorties du démultiplexeur.

EXEMPLE 9-15

Un technicien dépanne un système de transmission de données synchrone défectueux (figure 9-34). Avec un oscilloscope, il surveille les sorties du multiplexeur et du démultiplexeur pendant un cycle de transmission et il obtient les résultats reproduits à la figure 9-38. Quelles sont les causes possibles de ce dysfonctionnement?

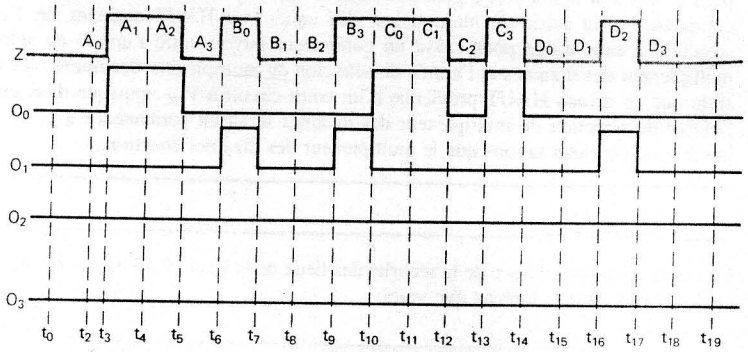


FIGURE 9-38 Exemple 9-14.

Solution L'examen des formes d'ondes observées met en lumière les points suivants:

- Tous les signaux semblent corrects entre t_0 et t_9, le signal O_0 écoulant les données série provenant du registre A, et le signal O_1 écoulant les données série provenant du registre B.
- Les sorties O_2 et O_3 ne sont jamais activées.
- Entre les instants t_{10} et t_{14}, la sortie O_0 écoule les données série du registre C, au lieu de la sortie O_2.
- De même, entre les instants t_{14} et t_{18}, c'est la sortie O_1 qui écoule les données série du registre D, au lieu de la sortie O_3.

Il semble que les entrées de sélection du démultiplexeur du récepteur ne choisissent que O_0 et O_1. Cela ne peut se produire que si S_1 est figé sur le niveau BAS. Ce maintien

d'un 0 sur ce noeud s'explique soit par un court-circuit interne en S_1 ou en Q_1, ou à l'entrée T du monostable, ou bien par un court-circuit externe à la masse. La position exacte de ce court-circuit peut être trouvée au moyen des techniques déjà exposées dans les chapitres précédents.

9-12 COMPARATEUR DE GRANDEURS

Un autre membre important de la catégorie des circuits MSI est le *comparateur de grandeurs*. Il s'agit d'un circuit logique combinatoire qui compare deux grandeurs binaires et produit des sorties qui désignent lequel des mots est le plus grand. La figure 9-39 illustre le symbole logique et la table de vérité du comparateur de grandeurs à 4 bits 74LS85, qui existe également dans les versions 7485 et 74HC85.

FIGURE 9-39 Symbole logique et table de vérité d'un comparateur de grandeurs 74LS85 (7485, 74HC85).

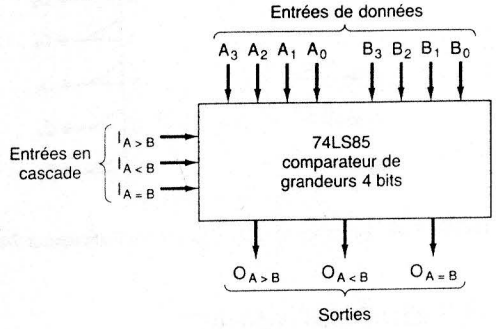


TABLE DE VÉRITÉ

COMPARAISON DES ENTRÉES				ENTRÉES EN CASCADE			SORTIES		
A_3, B_3	A_2, B_2	A_1, B_1	A_0, B_0	$I_{A>B}$	$I_{A<B}$	$I_{A=B}$	$O_{A>B}$	$O_{A<B}$	$O_{A=B}$
$A_3 > B_3$	X	X	X	X	X	X	H	B	B
$A_3 < B_3$	X	X	X	X	X	X	B	H	B
$A_3 = B_3$	$A_3 > B_2$	X	X	X	X	X	H	B	B
$A_3 = B_3$	$A_2 < B_2$	X	X	X	X	X	B	H	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 > B_1$	X	X	X	X	H	B	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 < B_1$	X	X	X	X	B	H	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 > B_0$	X	X	X	H	B	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 < B_0$	X	X	X	B	H	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	H	B	B	H	B	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	B	H	B	B	H	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	X	X	H	B	B	H
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	B	B	B	H	H	B
$A_3 = B_3$	$A_2 = B_2$	$A_1 = B_1$	$A_0 = B_0$	H	H	B	B	B	B

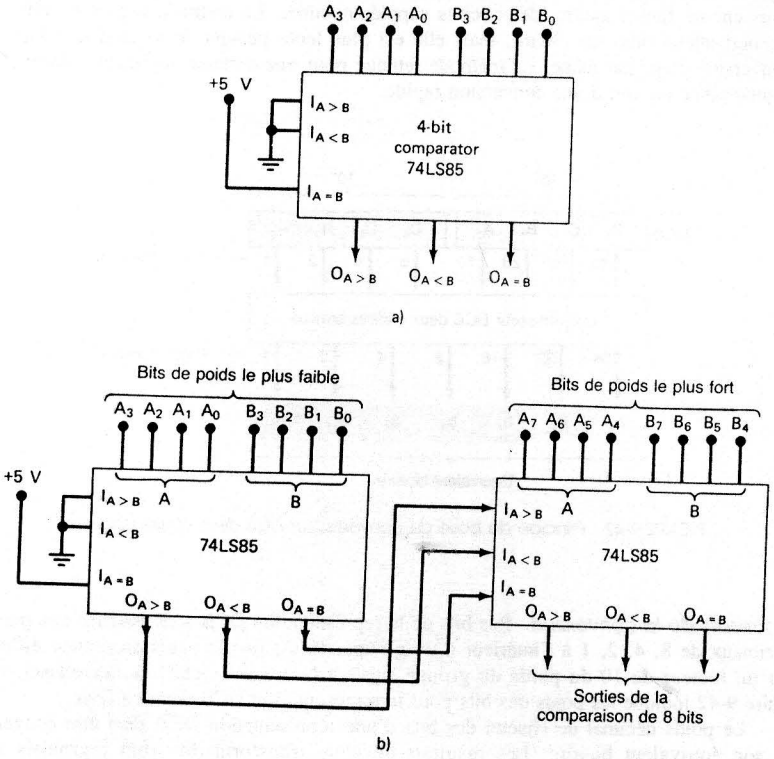
H = niveau HAUT
B = niveau BAS
X = indifférent

Entrées des données Le 74LS85 compare deux nombres binaires de 4 bits *non signés*. Le premier, placé sur $A_3A_2A_1A_0$, s'appelle le mot A; l'autre, placé sur $B_3B_2B_1B_0$, s'appelle le mot B. L'expression « mot » sert en technologie numérique à désigner un ensemble de bits véhiculant un certain genre d'information. Dans le cas présent, les mots A et B sont des grandeurs numériques.

Sorties Le 74LS85 possède trois sorties vraies au niveau HAUT; la sortie $O_{A>B}$ qui passe au niveau HAUT quand le mot A est plus grand que le mot B; la sortie $O_{A<B}$ qui passe au niveau HAUT quand le mot A est plus petit que le mot B; enfin, la sortie $O_{A=B}$ qui devient à son niveau vrai (HAUT) quand les deux mots ont la même grandeur.

Entrées en cascade Le montage en cascade des entrées constitue une façon d'étendre la comparaison à plus de 4 bits. Dans cette solution, on monte en cascade deux ou plusieurs comparateurs de 4 bits. Vous pouvez constater que les entrées en cascade ont les mêmes indices que les sorties correspondantes. Quand une comparaison de 4 bits est effectuée, comme à la figure 9-40 a), les entrées en cascade doivent être raccordées selon l'ordre indiqué afin d'obtenir du comparateur des sorties correctes.

FIGURE 9-40 a) Un 74LS85 câblé comme un comparateur 4 bits; b) deux 74LS85 montés en cascade afin de réaliser une comparaison de 8 bits.



Pour monter en cascade deux comparateurs, il faut que les sorties de celui de poids le plus faible soient connectées aux entrées ayant les mêmes indices dans le comparateur de rang supérieur. C'est ce que montre la figure 9-40 b), où le comparateur de gauche compare les 4 bits de poids inférieur de deux mots de 8 bits: $A_7A_6A_5A_4A_3A_2A_1A_0$ et $B_7B_6B_5B_4B_3B_2B_1B_0$. Les sorties sont amenées aux entrées en cascade du comparateur de droite, qui confronte déjà les 4 bits de poids le plus fort. Les sorties de ce dernier étage de comparaison contiennent le résultat final de la comparaison des mots de 8 bits.

EXEMPLE 9-16

Décrivez le fonctionnement du montage de comparaison de 8 bits illustré à la figure 9-40 b) pour les cas suivants:

a) $A_7A_6A_5A_4A_3A_2A_1A_0 = 10101111$; $B_7B_6B_5B_4B_3B_2B_1B_0 = 10110001$

b) $A_7A_6A_5A_4A_3A_2A_1A_0 = 10101111$; $B_7B_6B_5B_4B_3B_2B_1B_0 = 10101001$

Solution

a) Dans le comparateur de poids le plus fort, les entrées $A_7A_6A_5A_4 = 1010$ et $B_7B_6B_5B_4 = 1011$ sont évaluées et le résultat est $O_{A<B} = 1$, quels que soient les niveaux sur les entrées en cascade reportés du comparateur de poids le plus faible. En d'autres mots, dès que le comparateur de droite a établi la différence entre les bits des poids les plus forts, il sait quel mot de 8 bits est le plus grand et cela sans même connaître le résultat de la comparaison qui intervient dans le comparateur du rang inférieur.

b) Le comparateur de rang supérieur constate que $A_7A_6A_5A_4 = B_7B_6B_5B_4 = 1010$ et qu'il doit donc consulter les entrées en cascade pour connaître le résultat de la comparaison des bits de poids les plus faibles. Dans le comparateur de poids le plus faible, les entrées $A_3A_2A_1A_0 = 1111$ et $B_3B_2B_1B_0 = 1001$ sont évaluées, ce qui valide la sortie $O_{A>B}$. L'entrée en cascade correspondante $I_{A>B}$ prend la valeur 1. Le comparateur de poids le plus fort décèle cette valeur 1 et puisque ses propres entrées sont égales, il produit un niveau HAUT sur $O_{A>B}$, ce qui indique le résultat de la comparaison de 8 bits.

Applications Les comparateurs de grandeurs sont employés assez intensivement dans les circuits de décodage des adresses des ordinateurs; ce sont eux qui permettent de sélectionner le périphérique d'entrée/sortie ou de localiser la zone mémoire contenant les données que l'on veut retrouver. Ces éléments comparent le code d'adresse envoyé par le processeur central (UCT) à un code d'adresse matériel; si les deux coïncident, la sortie $O_{A=B}$ du comparateur active le dispositif ayant l'adresse correspondante. Les comparateurs de grandeurs sont également très utiles dans les applications de régulation où un nombre binaire figurant le comportement d'une variable physique régulée (comme la vitesse, la position) est comparé à une valeur de consigne. Les sorties du comparateur servent de déclencheur à l'envoi de signaux pour la conduite des mécanismes qui ramènent la variable physique vers son point de consigne. Nous examinons une application de comparateur type dans le problème 9-49.

Symbole IEEE/ANSI La figure 9-41 nous montre le symbole IEEE/ANSI du comparateur 74LS85. Notez l'emploi des lettres P et Q pour représenter les variables d'entrée. Il s'agit de la désignation sanctionnée par la norme IEEE/ANSI.

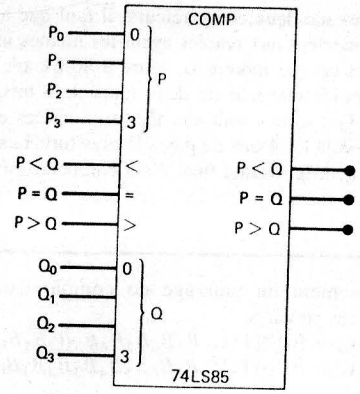


FIGURE 9-41 Le symbole IEEE/ANSI pour le comparateur.

QUESTIONS DE RÉVISION

1. Quel est la raison d'être des entrées en cascade du 74LS85?
2. Quelles sont les niveaux sur les sorties d'un 74LS85 si les différentes entrées sont: $A_3A_2A_1A_0 = B_3B_2B_1B_0 = 1001$, $I_{A>B} = I_{A<B} = 0$ et $I_{A=B} = 1$?

9-13 CONVERTISSEURS DE CODE

Un convertisseur de code est un circuit logique qui traduit les données exprimées dans un code binaire en un autre code binaire. Le pilote/décodeur DCB-7 segments déjà abordé est un convertisseur de code qui transforme un code d'entrée DCB en un code 7 segments indispensable pour allumer les DEL. Voici une liste partielle de certaines conversions de code fréquentes:

DCB à 7 segments
DCB à binaire
Binaire à DCB
Binaire à code Gray
Code Gray à binaire
ASCII à EBCDIC*
EBCDIC à ASCII
* EBCDIC est un code alphanumérique dû à IBM et semblable au code ASCII.

Comme exemple de convertisseur de code, nous retenons un convertisseur DCB-binaire. Avant de parler du circuit réel, revoyons la représentation DCB.

La suite des nombres décimaux à deux chiffres de 00 à 99 peut être représentée en DCB au moyen de deux groupes de quatre bits. Par exemple, 57_{10} s'écrit en DCB:

$$\overset{5}{\underbrace{0101}}\ \overset{7}{\underbrace{0111}} \quad \text{(DCB)}$$

La représentation binaire pure pour le nombre décimal 57 est:

$$57_{10} = 1111001_2$$

Le plus grand nombre décimal à deux chiffres, 99, a les représentations suivantes dans les deux codes:

$$99_{10} = 10011001 \text{ (DCB)} = 1100011_2$$

Observez la présence de seulement 7 bits dans la représentation binaire.

Principe de base Le schéma de la figure 9-42 nous illustre le principe de base d'un convertisseur DCB deux chiffres-binaire. Les entrées du convertisseur sont deux représentations codées de 4 bits: $D_0C_0B_0A_0$ qui est le chiffre pondéré à 10^0 ou le chiffre des unités, et $D_1C_1B_1A_1$ qui représente le chiffre des dizaines ou de poids 10^1. Les sorties du convertisseur sont $b_6b_5b_4b_3b_2b_1b_0$, soit les 7 bits de l'équivalent binaire de la valeur décimale d'entrée. Remarquez bien la différence des poids entre la représentation DCB et la représentation binaire.

Une utilisation habituelle du convertisseur DCB-binaire a lieu quand il faut transférer des données DCB, comme dans un multimètre, à un ordinateur de mémorisation ou de traitement. Les données doivent être converties en binaire afin que l'unité centrale, si elle n'est pas en mesure de travailler sur des données DCB, puisse les manipuler. La conversion DCB-binaire peut être accomplie soit par une solution matérielle, soit par une solution logicielle. La solution matérielle, que nous étudions ci-après, est généralement plus rapide mais encore faut-il ajouter des circuits supplémentaires. La méthode logicielle, elle, ne requiert aucun nouveau circuit, mais elle est plus lente puisque le logiciel doit faire la conversion étape par étape. La méthode retenue pour une certaine application dépend de l'importance ou non d'une conversion rapide.

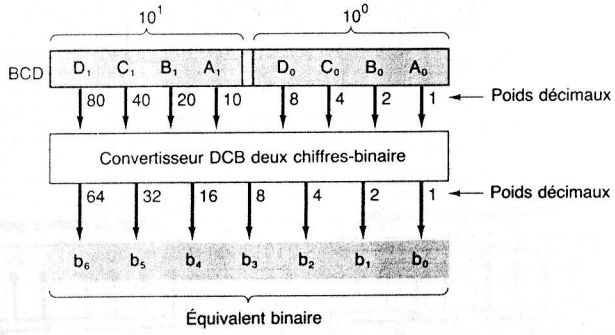


FIGURE 9-42 Principe de base du convertisseur DCB deux chiffres-binaire.

Processus de la conversion Les bits de la représentation DCB sont associés aux poids décimaux de 8, 4, 2, 1 à l'intérieur d'un groupe, tandis que le poids du groupe diffère par un facteur de 10 du poids du groupe suivant (entre deux chiffres décimaux). La figure 9-42 indique les poids des bits pour la représentation DCB deux chiffres.

Le poids décimal de chacun des bits d'une représentation DCB peut être converti en son équivalent binaire. Les résultats de cette transformation sont regroupés au

tableau 9-1. Au moyen de ces poids, on parvient à accomplir une conversion DCB-binaire de la manière suivante:

On obtient la somme des équivalents binaires de tous les bits qui sont à 1 dans la représentation DCB.

TABLEAU 9-1 ÉQUIVALENTS BINAIRES DES POIDS DÉCIMAUX DE CHAQUE BIT D'UNE REPRÉSENTATION DCB

Bit DCB	Poids décimal	b_6	b_5	b_4	b_3	b_2	b_1	b_0
A_0	1	0	0	0	0	0	0	1
B_0	2	0	0	0	0	0	1	0
C_0	4	0	0	0	0	1	0	0
D_0	8	0	0	0	1	0	0	0
A_1	10	0	0	0	1	0	1	0
B_1	20	0	0	1	0	1	0	0
C_1	40	0	1	0	1	0	0	0
D_1	80	1	0	1	0	0	0	0

EXEMPLE 9-17

Convertissez 01010010 (équivalent DCB du nombre décimal 52) en binaire. Même chose pour 10010101 (équivalent du nombre décimal 95).

Solution Écrivez les équivalents binaires de tous les 1 de la représentation DCB, puis additionnez-les (en binaire).

```
0 1 0 1 0 0 1 0        (DCB)
                 ──→   0000010   (2 en binaire)
                 ──→   0001010   (10 en binaire)
                 ──→ + 0101000   (40 en binaire)
                       ───────
                       0110100   (52 en binaire)

1 0 0 1 0 1 0 1        (DCB)
                 ──→   0000001   (1 en binaire)
                 ──→   0000100   (4 en binaire)
                 ──→   0001010   (10 en binaire)
                 ──→ + 1010000   (80 en binaire)
                       ───────
                       1011111   (95 en binaire)
```

Réalisation matérielle De toute évidence, une façon de réaliser le circuit logique d'un convertisseur est de recourir à des additionneurs binaires. La figure 9-43 nous fait voir deux 74LS83, des additionneurs parallèles de 4 bits, câblés de manière à effectuer la conversion. On voit là un montage parmi les nombreux montages qui réalisent la conversion.

Les deux additionneurs intégrés calculent la somme des bits DCB en respectant les équivalents binaires donnés au tableau 9-1. Par exemple, ce tableau indique qu'il n'y a que A_0 qui détermine la valeur du chiffre de poids le plus faible, b_0, de l'équivalent binaire. Et comme il n'y a pas de report sur ce rang binaire, A_0 est connecté directement sur b_0. Cette table nous apprend en outre que seuls les bits B_0 et A_1 contribuent à la valeur du

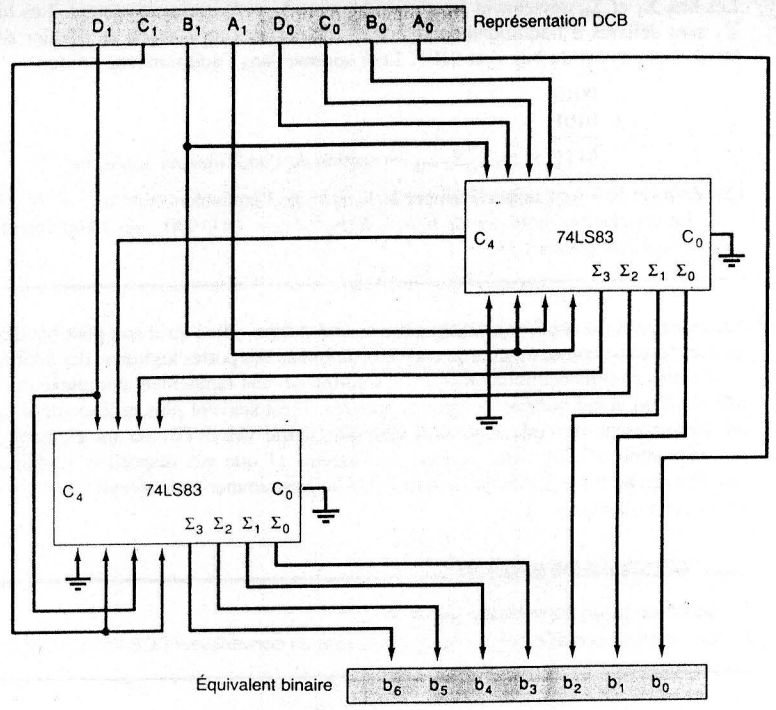


FIGURE 9-43 Convertisseur DCB-binaire réalisé avec des additionneurs parallèles 4 bits 74LS83.

bit b_1 du résultat binaire final. Ces deux bits sont donc combinés dans l'additionneur du haut pour donner la sortie b_1. De la même façon, seuls les bits DCB D_0, A_1 et C_1 interviennent dans la valeur du bit b_2. L'additionneur supérieur fait la somme de D_0 et A_1 pour générer le signal Σ_2, qui est amené dans l'additionneur du bas, qui l'additionne à C_1 afin d'obtenir b_3.

EXEMPLE 9-18

La représentation DCB du chiffre décimal 56 est appliquée à l'entrée du convertisseur de la figure 9-43. Trouvez les valeurs des sorties Σ pour chacun des additionneurs, de même que l'équivalent binaire final.

Solution Écrivez les bits de la représentation DCB 01010110 sur le schéma du circuit. Comme $A_0 = 0$, le bit b_0 de la sortie est 0.

Les entrées du haut de l'additionneur supérieur sont 0011. Les entrées du bas sont 0101. Cet additionneur fait la somme de ces entrées et produit:

$$\begin{array}{r} 0011 \\ +\ 0101 \\ \hline 1000 \end{array} = \Sigma_3\Sigma_2\Sigma_1\Sigma_0 \text{ les sorties de l'additionneur supérieur}$$

Les bits Σ_1 et Σ_0 deviennent les sorties binaires b_2 et b_1, respectivement. Les bits Σ_3 et Σ_2 sont délivrés à l'additionneur inférieur. Les entrées du haut de ce dernier sont alors 0010. Les entrées du bas sont 0101. Leur somme dans l'additionneur produit:

$$\begin{array}{r} 0010 \\ +\ 0101 \\ \hline 0111 \end{array} = \Sigma_3\Sigma_2\Sigma_1\Sigma_0 \text{ les sorties de l'additionneur inférieur}$$

Ces derniers bits sont respectivement $b_6b_5b_4b_3$ de l'équivalent binaire.

En conclusion, nous avons $b_6b_5b_4b_3b_2b_1b_0 = 0111000$, soit l'équivalent binaire exact du chiffre décimal 56.

Autres façons de réaliser un convertisseur de code Bien qu'il soit possible de réaliser tous les types de convertisseurs de code en combinant des portes logiques, des additionneurs ou d'autres circuits combinatoires, cette solution devient rapidement complexe et consommatrice d'un grand nombre de circuits intégrés. Il est souvent plus efficace de développer un convertisseur de code dans une mémoire morte (MORTE) ou un élément logique programmable (PLD). Nous verrons au chapitre 11 que ces dispositifs renferment des centaines de portes logiques qu'il est possible de programmer pour obtenir une large gamme de fonctions logiques.

QUESTIONS DE RÉVISION

1. Qu'est-ce qu'un convertisseur de code?
2. Combien de sorties binaires aurons-nous dans un convertisseur DCB trois chiffres-binaire?

9-14 REGISTRES TROIS ÉTATS

Dans la plupart des ordinateurs d'aujourd'hui, particulièrement dans les micro-ordinateurs, la transmission des données s'accomplit par l'entremise d'un faisceau commun de connecteurs appelé un **bus de donnée**. Dans les ordinateurs, de nombreux dispositifs ont leurs sorties et leurs entrées reliées aux conducteurs communs du bus. À cause de cela, les sorties de ces dispositifs sont souvent des sorties trois états.

Les dispositifs habituellement raccordés sur un bus de donnée sont: les microprocesseurs (chapitre 12), les puces mémoires à semiconducteurs que nous étudierons au chapitre 11 ainsi que les convertisseurs numériques-analogiques et analogiques-numériques qui seront traités au chapitre 10.

Dans presque tous les cas, les dispositifs raccordés à un bus contiennent des registres qui conservent leurs données. Les sorties de ces registres possèdent des tampons trois états afin de permettre la transmission des données au moyen d'un bus. Il existe de nombreux registres en version CI qui comprennent déjà des tampons trois états dans la puce même. L'un de ceux-ci est le TTL 74173 (et ses équivalents 74LS173 et 74HC173); son schéma logique et sa table de vérité sont reproduits à la figure 9-44.

Le 74173 est un registre de 4 bits à écriture parallèle et lecture parallèle. Notez que les sorties des bascules aboutissent à des tampons trois états qui fournissent les sorties O_0–O_3. Remarquez également que les entrées de données D_0–D_3 sont connectées aux entrées D des bascules du registre par l'intermédiaire de portes logiques. Ces portes

Table de vérité

entrées					sortie
MR	CP	$\overline{IE}_1$	$\overline{IE}_1$	D_n	Q
H	X	X	X	X	B
B	B	X	X	X	Q_0
B	∫	H	X	X	Q_0
B	∫	X	H	X	Q_0
B	∫	B	B	B	B
B	∫	B	B	H	H

Quand $\overline{OE}_1$ ou $\overline{OE}_2$ est au niveau HAUT, la sortie est DÉCONNECTÉE (haute impédance); cependant, ceci ne modifie en rien le contenu ou le fonctionnement séquentiel du registre

H = niveau de tension HAUT
B = niveau de tension BAS
X = indifférent
Q_0 = sortie avant le FM

schéma logique

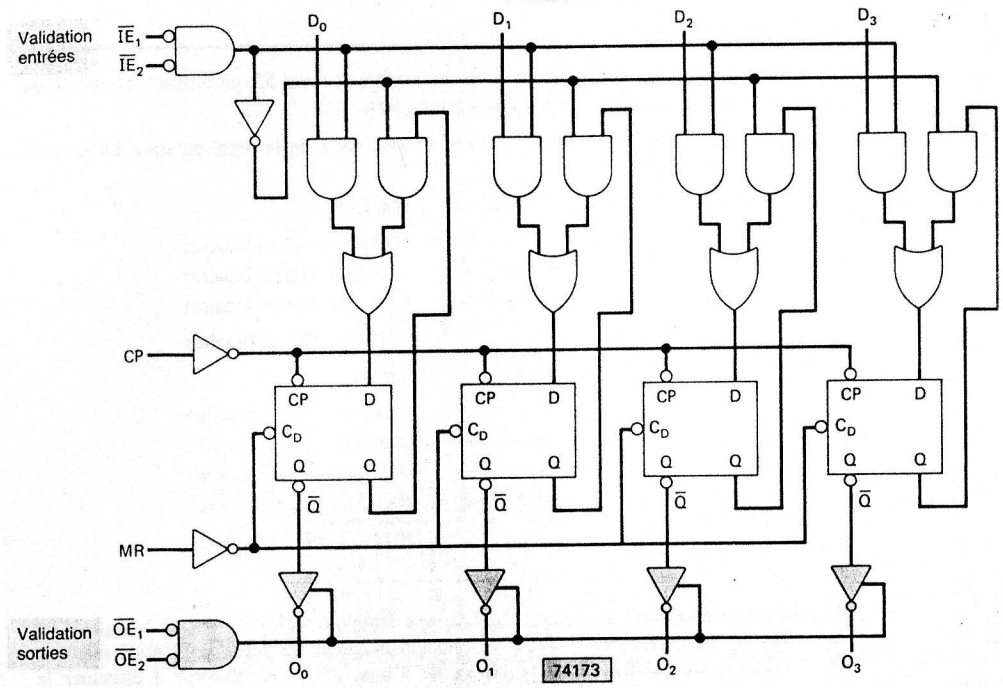


FIGURE 9-44 Table de vérité et schéma logique du registre trois états 74173. (Gracieuseté de Fairchild, filiale Schlumberger).

permettent de choisir entre deux modes de fonctionnements: 1) *chargement*, dans ce mode de fonctionnement les données sur les entrées D_0–D_3 sont transférées dans les bascules au moment du FM de l'impulsion d'horloge en *CP*; 2) *mémorisation*, dans ce mode de fonctionnement les données placées dans le registre demeurent inchangées au moment du passage du FM qui arrive à la borne *CP*.

EXEMPLE 9-19

a) Dites quelles sont les conditions d'entrée qui choisissent le chargement.
b) Dites quelles sont les conditions d'entrée qui choisissent la mémorisation.
c) Dites quelles sont les conditions d'entrée qui autorisent le transfert des sorties du registre interne sur O_0 à O_3.

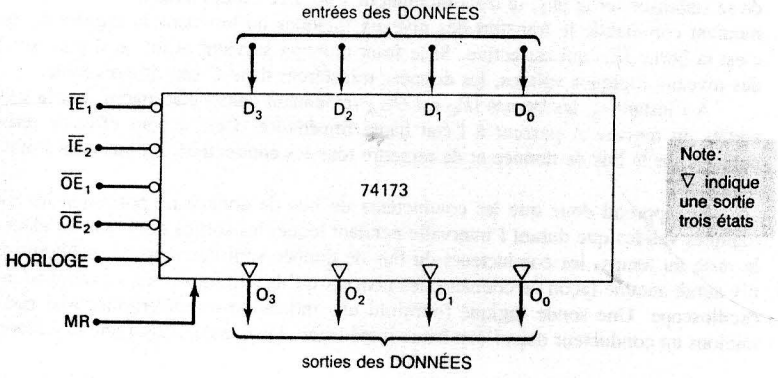
Solution

a) Les deux dernières entrées de la table de vérité démontrent que chaque borne Q prend la valeur présente sur son entrée D à l'instant d'un front montant en *CP*, à la condition que *MR* soit au niveau BAS et que les deux entrées de validation, $\overline{IE}_1$ et $\overline{IE}_2$, soient aussi au niveau BAS.
b) La troisième et la quatrième ligne de la table de vérité établissent que lorsque l'une ou l'autre des entrées $\overline{IE}$ est HAUTE, les entrées D n'ont aucun effet sur les sorties Q, qui alors ne changent pas de valeur au passage du front montant.
c) Les tampons de sortie sont validés quand les deux entrées de validation des sorties, $\overline{OE}_1$ et $\overline{OE}$, sont BASSES. Cela transfère les sorties du registre jusque sur les sorties O_0 à O_3. Si l'une ou l'autre des entrées de validation des sorties est HAUTE, les tampons sont bloqués et les sorties sont dans leur état de haute impédance.

Vous pouvez constater que les sorties $\overline{OE}$ n'affectent d'aucune manière l'opération de chargement des données. Elles ont essentiellement pour rôle d'ouvrir ou de fermer le passage des sorties du registre vers les sorties extérieures.

La figure 9-45 nous fait voir le symbole logique du 74173. C'est ce symbole que nous utiliserons dans la section suivante traitant des bus de donnée.

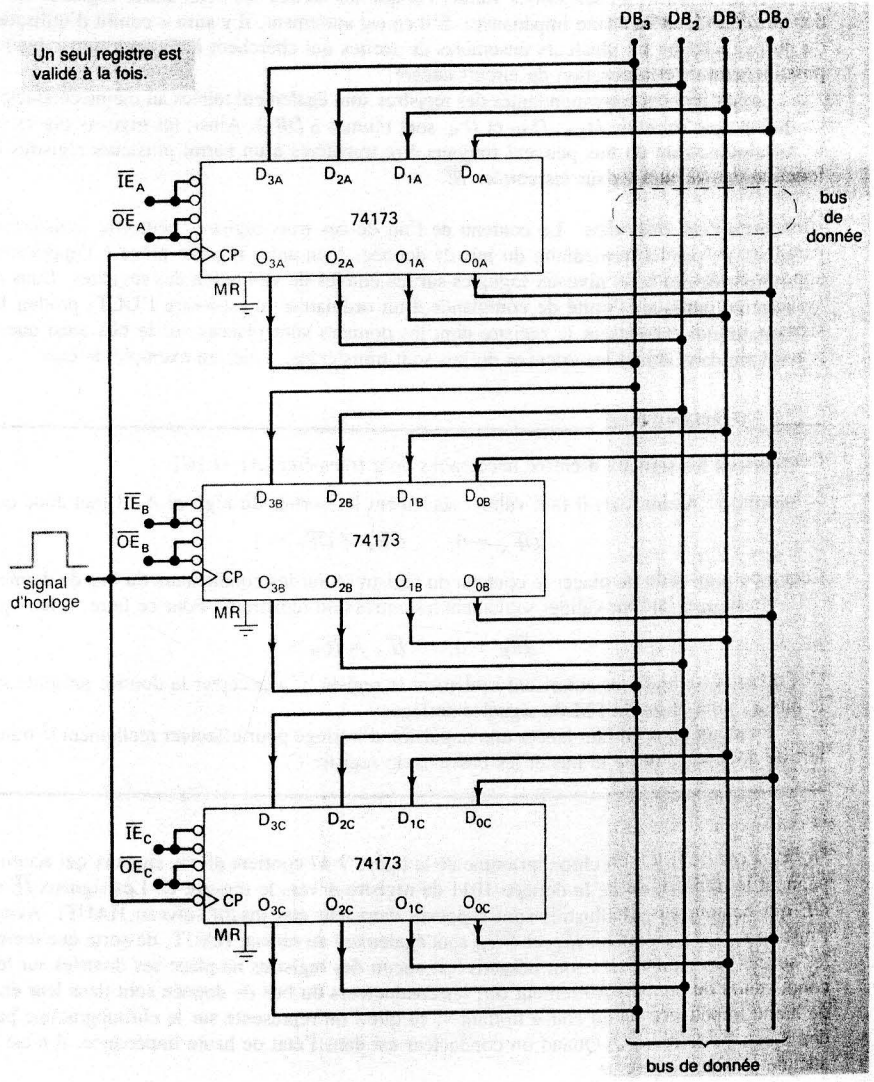
FIGURE 9-45 Symbole logique du 74173.



9-15 BUS DE DONNÉE

Le bus de donnée est un élément d'une grande importance dans les ordinateurs, importance que vous serez en mesure d'apprécier lorsque vous étudierez plus tard les mémoires et les microprocesseurs. Pour l'instant, nous nous contenterons d'illustrer le concept du bus de donnée dans le cas des transferts de données registre à registre. Sur la figure 9-46, on a

FIGURE 9-46 Registres trois états raccordés à un bus de donnée.



schématisé un système à bus formé de trois registres trois états 74173. Il convient de remarquer que pour chaque registre on a réuni ensemble les deux entrées $\overline{OE}$, de même que les deux entrées $\overline{IE}$ afin de n'avoir qu'une entrée $\overline{OE}$ et une entrée $\overline{IE}$. Il faut également noter que les registres sont appelés, en partant d'en haut, registres *A*, *B* et *C*. D'ailleurs, **toutes les entrées et toutes les sorties sont caractérisées par des indices correspondants.**

Dans ce montage, le bus de donnée est formé de quatre conducteurs appelés DB_0–DB_3. **Les sorties correspondantes de chaque registre sont connectées au même conducteur du bus** (O_{3A}, O_{3B} et O_{3C} sont branchés sur DB_3). Étant donné que des sorties de ces trois registres sont connectées ensemble, il est indispensable que seulement un des registres à la fois ait ses sorties validées et que les sorties des deux autres registres soient déconnectées (état haute impédance). S'il en est autrement, il y aura « conflit d'utilisation du bus » (deux ou plusieurs ensembles de sorties qui cherchent à s'approprier le bus) et possiblement détérioration du circuit intégré.

Les entrées correspondantes des registres sont également reliées au même conducteur du bus (par exemple D_{3A}, D_{3B} et D_{3C} sont réunies à DB_3). Ainsi, les niveaux placés sur les conducteurs du bus peuvent toujours être transférés à un parmi plusieurs registres en fonction des valeurs sur les entrées $\overline{IE}$.

Transfert de données Le contenu de l'un de ces trois registres peut être transféré en parallèle, par l'intermédiaire du bus de donnée, à un autre registre grâce à l'application appropriée des bons niveaux logiques sur les entrées de validation des registres. Dans un système pratique, l'unité de commande d'un ordinateur (c'est-à-dire l'UCT) produit les signaux qui choisissent le registre dont les données sont placées sur le bus ainsi que le registre dans lequel les données du bus sont transférées. Voici un exemple de ceci.

EXEMPLE 9-20

Décrivez les signaux d'entrée nécessaires pour transférer $[A] \rightarrow [C]$.

Solution Avant tout, il faut valider seulement les sorties du registre *A*. Il faut donc que

$$\overline{OE}_A = 0, \qquad \overline{OE}_B = \overline{OE}_C = 1$$

Ceci a pour effet de placer le contenu du registre *A* sur les conducteurs du bus de donnée.

Ensuite, il faut valider seulement les entrées du registre *C*. Pour ce faire, il faut que

$$\overline{IE}_C = 0, \qquad \overline{IE}_A = \overline{IE}_B = 1$$

Ces niveaux logiques autorisent seulement le registre *C* à accepter la donnée présente sur le bus à l'arrivée du FM du signal d'horloge.

Finalement, il faut lancer une impulsion d'horloge pour effectuer réellement le transfert de donnée entre le bus et les bascules du registre *C*.

Signaux de bus Le chronogramme de la figure 9-47 contient divers signaux qui accompagnent le transfert de la donnée 1011 du registre *A* vers le registre *C*. Les signaux $\overline{IE}$ et $\overline{OE}$ qui ne sont pas illustrés sont supposés dans leur état inactif (niveau HAUT). Avant l'instant t_1, les signaux $\overline{IE}_C$ et $\overline{OE}_A$ sont également au niveau HAUT, de sorte que toutes les sorties des registres sont désactivées: aucun des registres ne place ses données sur les conducteurs du bus. Autrement dit, les conducteurs du bus de donnée sont dans leur état de haute impédance ou état « flottant », ce que l'on représente sur le chronogramme par des lignes hachurées. Quand un conducteur est dans l'état de haute impédance, il n'est à aucune tension précise.

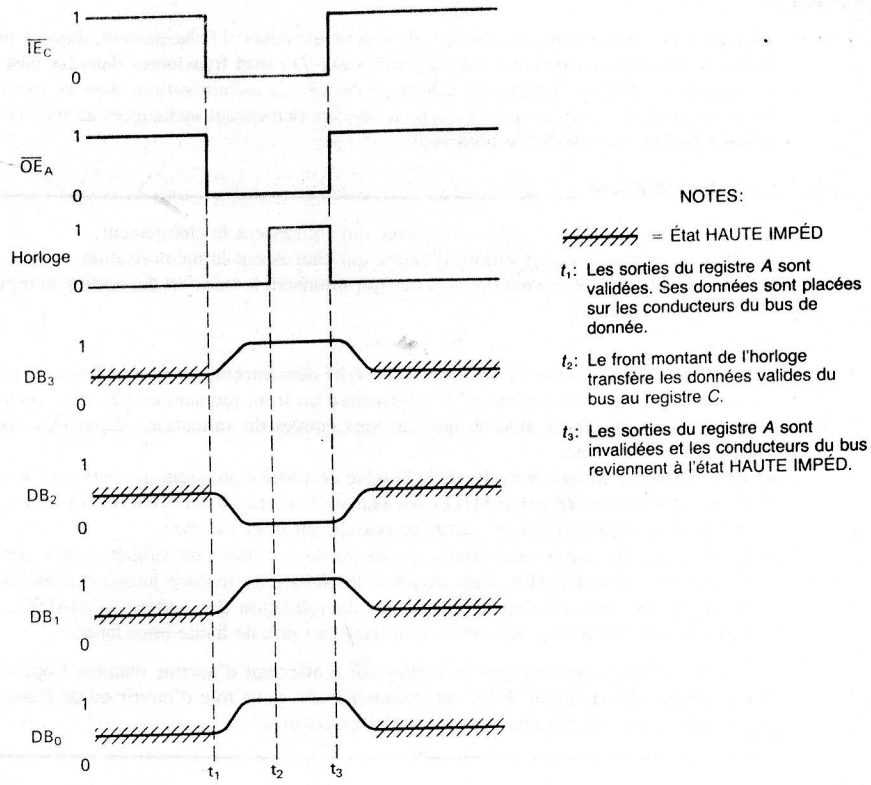


FIGURE 9-47 Évolution des signaux durant le transfert de la donnée 1011 du registre *A* au registre *C*.

À l'instant t_1, les entrées $\overline{IE}_C$ et $\overline{OE}_A$ sont activées. Les sorties du registre *A* sont validées et dès lors les conducteurs DB_3 à DB_0 du bus de donnée commencent à passer de l'état haute impédance à l'état 1011. Après un certain temps, qui permet aux niveaux de se stabiliser sur le bus, le front montant de l'horloge est appliqué à l'instant t_2. Ce front montant commande le transfert des niveaux logiques du bus dans le registre *C*, puisque c'est la borne $\overline{IE}_C$ qui est active. Si le front montant survient avant qu'il y ait sur le bus des niveaux logiques valides, les données transférées dans *C* sont imprévisibles.

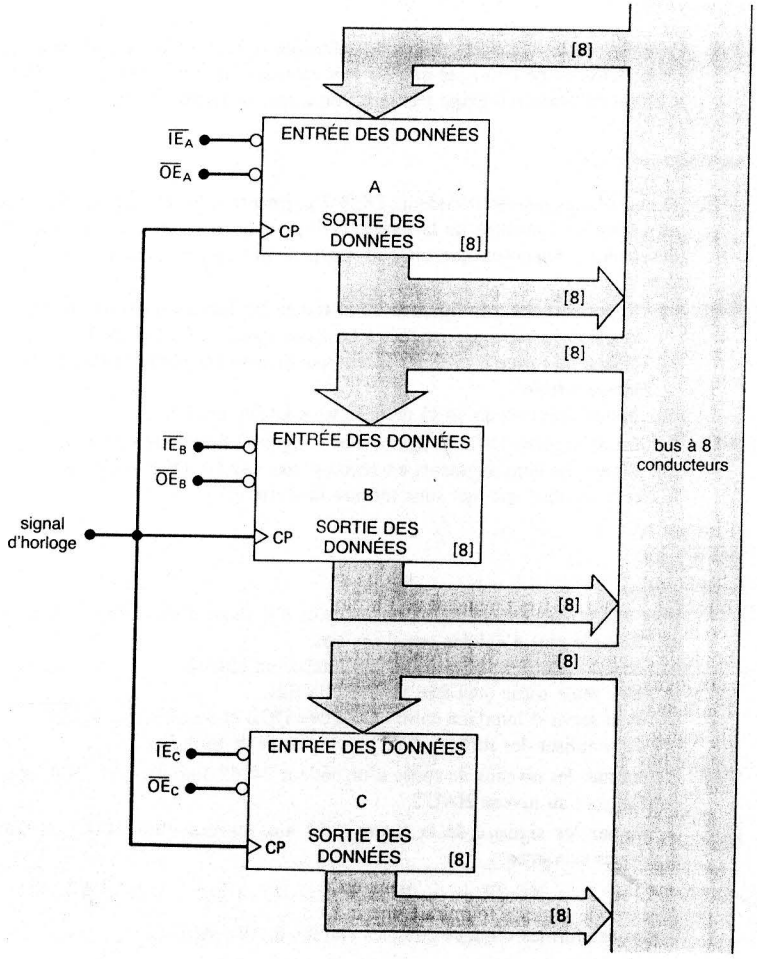
À l'instant t_3, les bornes $\overline{IE}_C$ est $\overline{OE}_A$ reviennent dans l'état inactif. Par la suite, les sorties du registre *A* passent à l'état haute impédance. Cela a pour effet de retirer les données sur le bus de donnée et de remettre tous les conducteurs du bus dans l'état haute impédance.

On apprend donc que les conducteurs du bus de donnée ne possèdent les niveaux logiques valides que durant l'intervalle pendant lequel les sorties de *A* sont validées. Tout le reste du temps, les conducteurs du bus de donnée sont dans un état indéterminé et il n'y aurait aucune façon de connaître les niveaux qu'ils portent si on les examinait avec un oscilloscope. Une sonde logique fournirait une indication « indéterminée » si nous examinions un conducteur dans l'état haute impédance. On constate aussi que la variation des

signaux sur le bus est relativement lente. Bien que cette lenteur ait été exagérée sur le chronogramme, c'est une caractéristique bien connue des bus et elle est due à la charge capacitive de chaque conducteur. Cette charge est la somme de la capacité parasite et des capacités associées à chaque connexion avec le conducteur (pour les entrées et les sorties).

Expansion du bus Il est aisé de voir comment on peut augmenter le nombre de registres raccordés au bus et, partant, accroître la quantité de transferts interregistres effectués par le biais du bus. Évidemment, pour chaque nouveau registre, s'ajoutent deux nouvelles entrées de validation qui contrôle le transfert des données. En plus, s'ajoutent pour chaque registre les entrées de données qui sont pilotées par les sorties du registre qui procédent à une opération de transfert sur le bus. Dans les systèmes dotés d'un grand nombre de

FIGURE 9-48 Représentation simplifiée des bus.



registres (ou dispositifs), tous raccordés au bus, les sorties des registres peuvent aboutir à un tampon appelé un **pilote de bus**. Même quand tous les dispositifs sont de la famille MOS ou CMOS, la charge capacitive totale associée à chaque conducteur parvient à dégrader considérablement les temps de transition des changements de niveau sur le conducteur. Un pilote de bus intégré sur puce a une résistance de sortie très faible pour chaque niveau logique, de sorte que la charge et la décharge des condensateurs de bus se font plus rapidement que dans le cas d'un CI normal.

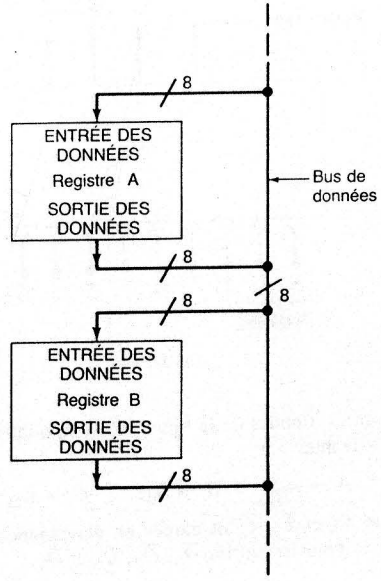
Reportez-vous aux problèmes 9-56 à 9-58 où vous trouverez l'exposition d'un cas réel d'extension de bus, de même qu'une discussion des circuits qui sélectionnent les dispositifs envoyant et recevant les données.

Représentation simplifiée des bus Généralement, de nombreux dispositifs sont branchés sur le même bus de donnée. Sur un schéma de circuit, il peut devenir difficile de s'y reconnaître étant donné le grand nombre de conducteurs et de connexions. C'est la raison pour laquelle on utilise presque toujours une représentation simplifiée pour les connexions au bus de donnée sur les schémas fonctionnels et sur tous les autres schémas, sauf les plus détaillés. On peut voir un exemple de cette représentation simplifiée à la figure 9-48 dans le cas d'un bus de donnée à huit conducteurs.

Les lignes qui quittent le bus de donnée ou qui aboutissent à ce dernier sont représentées par de larges flèches. Les chiffres placés entre crochets indiquent le nombre de bits que renferme chaque registre de même que le nombre de conducteurs qui connectent les entrées et les sorties des registres au bus.

Une représentation encore plus simple d'un bus qui est en voie de devenir une norme, est reproduite à la figure 9-49. Il s'agit d'un trait unique pour figurer le bus et les connexions

FIGURE 9-49 Représentation simplifiée d'un bus: on utilise un seul trait pour figurer tout le faisceau. Le « /8 » indique un bus à 8 conducteurs.



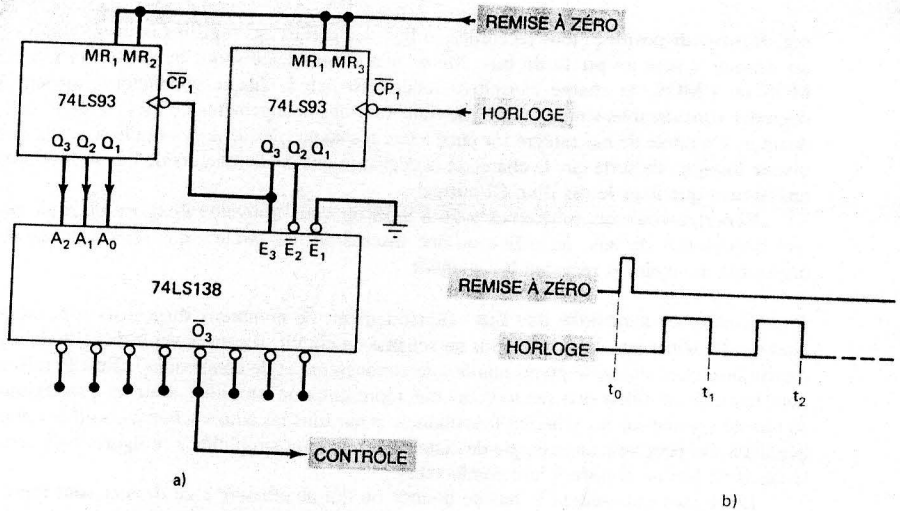


FIGURE 9-52

9-7 Le décodeur 7442 de la figure 9-5 ne possède pas d'entrée VALIDATION. Il est toutefois possible de s'en servir comme décodeur 1 parmi 8 si on n'utilise pas les sorties $\overline{O}_8$ et $\overline{O}_9$ et qu'on se sert de l'entrée D comme entrée de VALIDATION. C'est ce que nous montre la figure 9-53. Prouvez que ce montage fonctionne comme un décodeur 1 parmi 8 validé et montrez comment les niveaux sur D valident ou invalident les sorties.

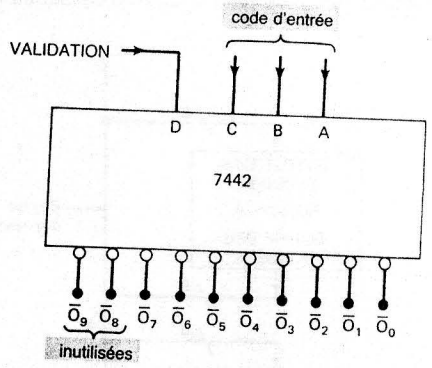


FIGURE 9-53

9-8 Soient les formes d'ondes de la figure 9-54. Appliquez ces signaux à un 74LS138 de la façon suivante:

$$A \rightarrow A_0; \qquad B \rightarrow A_1; \qquad C \rightarrow A_2; \qquad D \rightarrow E_3$$

Supposez que $\overline{E}_1$ et $\overline{E}_2$ soient gardés en permanence au niveau BAS; tracez les formes d'ondes pour les sorties $\overline{O}_0$, $\overline{O}_3$, $\overline{O}_6$ et $\overline{O}_7$.

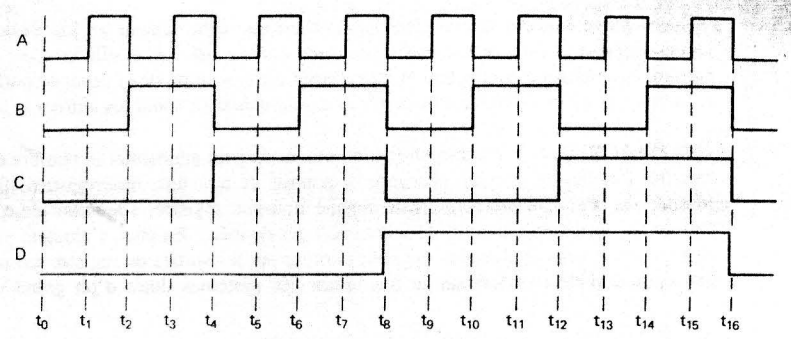


FIGURE 9-54

9-9 Modifiez le circuit de la figure 9-6 afin que le relais K_1 reste en mode de travail (sous tension) de t_3 à t_5 et que K_2 soit en mode de travail de t_6 à t_9. (*Suggestion:* Cette modification n'exige pas que l'on ajoute d'autres circuits.)

SECTIONS 9-2 et 9-3

9-10 Raccordez les pilotes/décodeurs DCB-7 segments et les afficheurs DEL-7 segments au circuit de l'horloge de la figure 7-42 pour qu'il y ait affichage des minutes et des heures. Supposez que chaque segment tire approximativement 10 mA, sous 2,5 V.

9-11 a) Reportez-vous à la figure 9-10 et tracez les formes d'ondes du segment et de la plaque arrière par rapport à la masse quand COMMANDE = 0. Dessinez ensuite la forme d'onde de la tension du segment par rapport à la tension de la plaque arrière.

b) Même question qu'en a) mais lorsque COMMANDE = 1.

9-12 Le décodeur/pilote DCB-7 segments de la figure 9-8 contient tout le montage logique qui allume les bons segments en réponse aux entrées DCB appropriées. Concevez le circuit logique qui met sous tension la sortie g.

SECTION 9-4

9-13 CONTRÔLE

Pour chacun des énoncés qui suit, indiquez s'il s'agit d'un codeur ou d'un décodeur.

a) Possède plus d'entrées que de sorties.
b) Sert à traduire l'enfoncement de touches en binaire.
c) Une seule sortie peut être activée à la fois.
d) Peut servir d'interface entre une entrée DCB et un afficheur à DEL.
e) Peut utiliser des sorties pilote pour soutenir de forts *I* et *V*.

9-14 Déterminez les niveaux de sortie d'un codeur 74147 dont $\overline{A}_8 = \overline{A}_4 = 0$, et les autres entrées sont au niveau HAUT.

9-15 Appliquez les signaux de la figure 9-54 aux entrées d'un 74147 en suivant les directives suivantes:

$$A \rightarrow \overline{A}_7; \qquad B \rightarrow \overline{A}_4; \qquad C \rightarrow \overline{A}_2; \qquad D \rightarrow \overline{A}_1$$

Tracez les formes d'ondes pour les entrées du décodeur.

à ce bus. Une barre oblique (/) intercepte ce trait, suivie d'un nombre correspondant au nombre de conducteurs parallèles formant le bus.

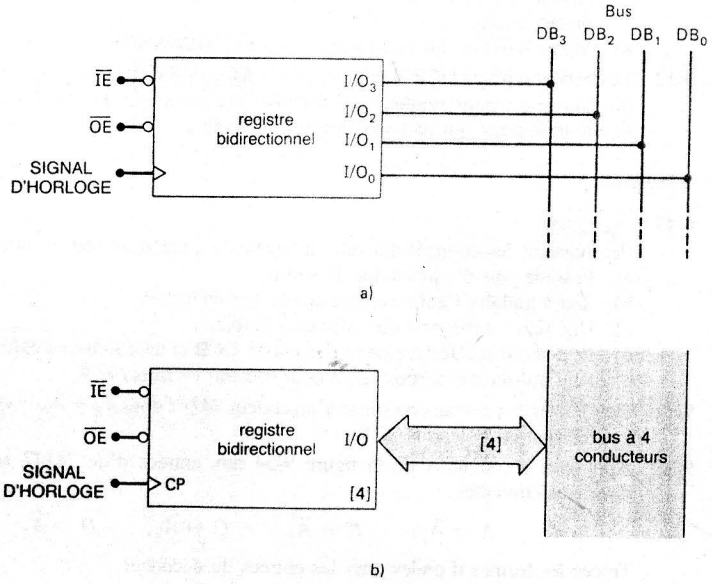
Bus bidirectionnel Chacun des registres de la figure 9-46 possède des entrées et des sorties raccordées au bus de donnée, de sorte que les entrées et les sorties aboutissant au même conducteur sont en court-circuit les unes avec les autres. Par exemple, la sortie O_2 de chaque registre est raccordée à l'entrée D_2 en raison de leur branchement commun sur DB_2. Évidemment, ce n'est pas cela qui se passe vraiment si on a intercalé des pilotes entre les sorties et le bus.

Parce que des entrées et des sorties sont souvent raccordées au même bus, les fabricants ont mis au point des circuits intégrés dans lesquels les entrées et les sorties sont déjà raccordées *dans la puce* afin de réduire le nombre de broches. La figure 9-50 nous montre un registre de 4 bits de ce genre. Les lignes d'entrée (D_0–D_3) et les lignes de sortie (O_0–O_3) distinctes sont remplacées par des lignes entrée/sortie (I/O_0–I/O_3).

Chaque ligne d'entrée/sortie fonctionne soit comme une entrée ou comme une sortie selon les états placés sur les entrées de validation. On parle alors dans ce cas de **lignes de données bidirectionnelles**. Le 74LS299 est un registre de 8 bits doté de lignes I/O communes. On retrouve dans de nombreuses puces mémoires et dans des microprocesseurs cette fonction de transfert bidirectionnel des données.

Nous reviendrons sur le sujet crucial des bus de donnée au moment du traitement détaillé des systèmes de mémoire au chapitre 11, de même qu'au chapitre 12 consacré à une introduction aux microprocesseurs et micro-ordinateurs.

FIGURE 9-50 a) Registre bidirectionnel; b) schéma simplifié.



QUESTIONS DE RÉVISION

1. Définissez le terme « bus de donnée ».
2. Définissez « conflit d'utilisation du bus » et dites ce qu'il faut faire pour le prévenir.
3. Indiquez la fonction d'un pilote de bus.
4. Indiquez les raisons pour lesquelles il est commode d'avoir des lignes d'entrée/sortie communes.
5. Redessinez la figure 9-50a en adhérant à la représentation monotrait. La réponse est la figure 9-51.

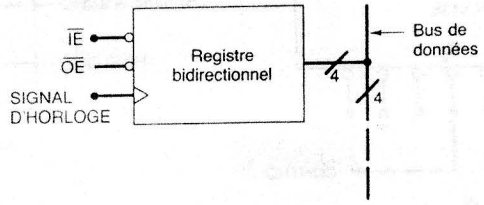


FIGURE 9-51

PROBLÈMES

SECTION 9-1

9-1 Reportez-vous à la figure 9-3. Déterminez le niveau de chacune des sorties du décodeur correspondant aux ensembles de conditions sur les entrées que voici.
   a) Toutes les entrées sont BASSES.
   b) Toutes les entrées sont BASSES, sauf E_3 = HAUT.
   c) Toutes les entrées sont HAUTES, sauf $\overline{E}_1 = \overline{E}_2$ = BAS.
   d) Toutes les entrées sont HAUTES.

9-2 Donnez le nombre de voies d'entrée et de sortie d'un décodeur sur l'entrée duquel on peut placer 64 combinaisons différentes.

9-3 Indiquez l'ensemble des conditions d'entrée qui donne un niveau BAS sur $\overline{O}_6$ d'un 74LS138.

9-4 Montrez comment agencer des 74LS138 pour obtenir un décodeur 1 parmi 16.

9-5 La figure 9-52 montre comment on utilise un décodeur pour produire des signaux de contrôle. Supposez que l'impulsion REMISE À ZÉRO soit produite à l'instant t_0 et déterminez la forme d'onde de CONTRÔLE pour 32 impulsions.

9-6 Modifiez le circuit de la figure 9-52 pour produire une forme d'onde de CONTRÔLE qui soit au niveau BAS entre t_{20} et t_{24}. (*Suggestion:* cette modification n'exige aucun circuit logique supplémentaire.)

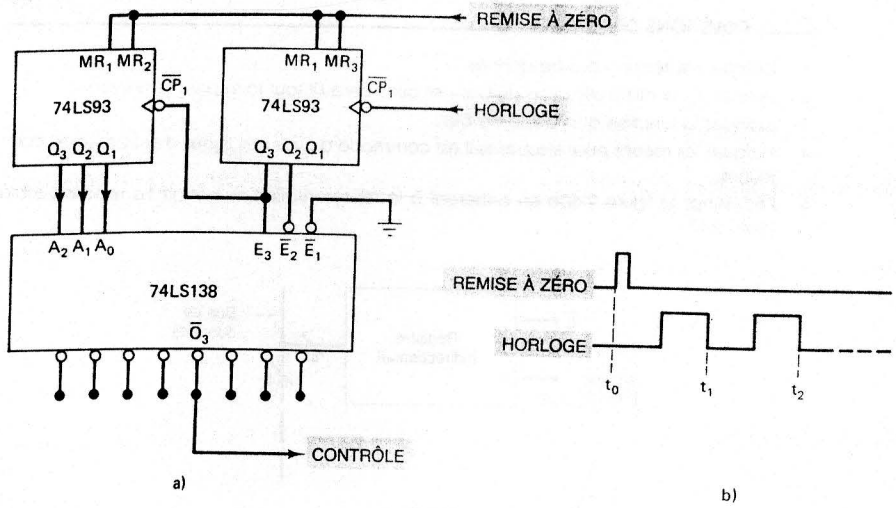


FIGURE 9-52

9-7 Le décodeur 7442 de la figure 9-5 ne possède pas d'entrée VALIDATION. Il est toutefois possible de s'en servir comme décodeur 1 parmi 8 si on n'utilise pas les sorties $\overline{O}_8$ et $\overline{O}_9$ et qu'on se sert de l'entrée D comme entrée de VALIDATION. C'est ce que nous montre la figure 9-53. Prouvez que ce montage fonctionne comme un décodeur 1 parmi 8 validé et montrez comment les niveaux sur D valident ou invalident les sorties.

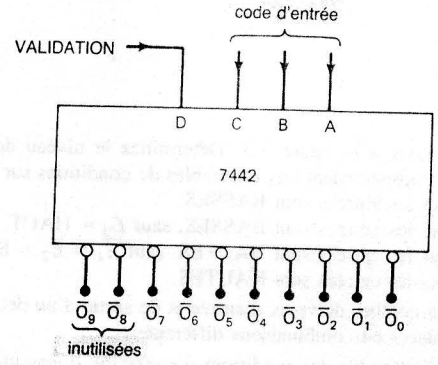


FIGURE 9-53

9-8 Soient les formes d'ondes de la figure 9-54. Appliquez ces signaux à un 74LS138 de la façon suivante:

$$A \rightarrow A_0; \qquad B \rightarrow A_1; \qquad C \rightarrow A_2; \qquad D \rightarrow E_3$$

Supposez que $\overline{E}_1$ et $\overline{E}_2$ soient gardés en permanence au niveau BAS; tracez les formes d'ondes pour les sorties $\overline{O}_0$, $\overline{O}_3$, $\overline{O}_6$ et $\overline{O}_7$.

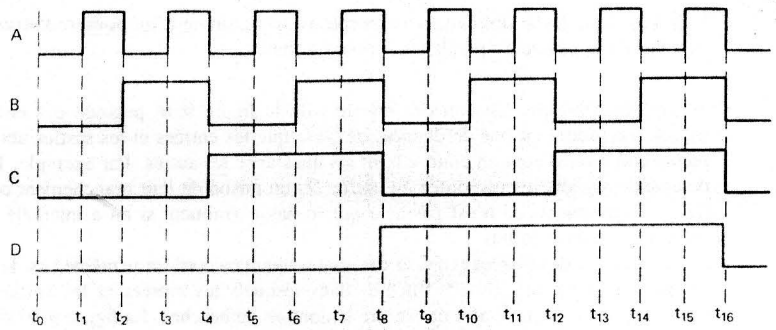


FIGURE 9-54

9-9 Modifiez le circuit de la figure 9-6 afin que le relais K_1 reste en mode de travail (sous tension) de t_3 à t_5 et que K_2 soit en mode de travail de t_6 à t_9. (*Suggestion:* Cette modification n'exige pas que l'on ajoute d'autres circuits.)

SECTIONS 9-2 et 9-3

9-10 Raccordez les pilotes/décodeurs DCB-7 segments et les afficheurs DEL-7 segments au circuit de l'horloge de la figure 7-42 pour qu'il y ait affichage des minutes et des heures. Supposez que chaque segment tire approximativement 10 mA, sous 2,5 V.

9-11 a) Reportez-vous à la figure 9-10 et tracez les formes d'ondes du segment et de la plaque arrière par rapport à la masse quand COMMANDE = 0. Dessinez ensuite la forme d'onde de la tension du segment par rapport à la tension de la plaque arrière.

b) Même question qu'en a) mais lorsque COMMANDE = 1.

9-12 Le décodeur/pilote DCB-7 segments de la figure 9-8 contient tout le montage logique qui allume les bons segments en réponse aux entrées DCB appropriées. Concevez le circuit logique qui met sous tension la sortie g.

SECTION 9-4

9-13 CONTRÔLE

Pour chacun des énoncés qui suit, indiquez s'il s'agit d'un codeur ou d'un décodeur.

a) Possède plus d'entrées que de sorties.
b) Sert à traduire l'enfoncement de touches en binaire.
c) Une seule sortie peut être activée à la fois.
d) Peut servir d'interface entre une entrée DCB et un afficheur à DEL.
e) Peut utiliser des sorties pilote pour soutenir de forts I et V.

9-14 Déterminez les niveaux de sortie d'un codeur 74147 dont $\overline{A}_8 = \overline{A}_4 = 0$, et les autres entrées sont au niveau HAUT.

9-15 Appliquez les signaux de la figure 9-54 aux entrées d'un 74147 en suivant les directives suivantes:

$$A \rightarrow \overline{A}_7; \qquad B \rightarrow \overline{A}_4; \qquad C \rightarrow \overline{A}_2; \qquad D \rightarrow \overline{A}_1$$

Tracez les formes d'ondes pour les entrées du décodeur.

9-16 La figure 9-55 reproduit le schéma fonctionnel d'un circuit logique servant à contrôler le nombre d'exemplaires faits par une photocopieuse. L'opérateur choisit le nombre d'exemplaires désiré en fermant l'un des interrupteurs de sélection S_1–S_9. Ce nombre est pris en charge par le codeur qui le transforme en un nombre DCB, puis il est envoyé au circuit comparateur. L'opérateur appuie alors momentanément sur le bouton DÉBUT, ce qui a pour effet d'initialiser le compteur et de mettre à 1 la sortie OPÉRATION; cette dernière sortie commande à la machine de faire les reproductions. Pendant que la machine reproduit, une impulsion copie est produite et fournie au compteur DCB. Les sorties de ce compteur sont comparées sans arrêt dans le comparateur aux sorties du codeur d'interrupteurs. Quand les deux nombres DCB sont identiques, ce qui indique que le nombre d'exemplaires voulu est atteint, la sortie $\overline{X}$ du comparateur passe au niveau BAS; ceci ramène le signal opération au niveau BAS et arrête la reproduction d'exemplaires par la machine. Si on appuie à nouveau sur le bouton DÉBUT, ce processus recommence. Concevez tous les circuits logiques du comparateur et de la section de commande de ce système.

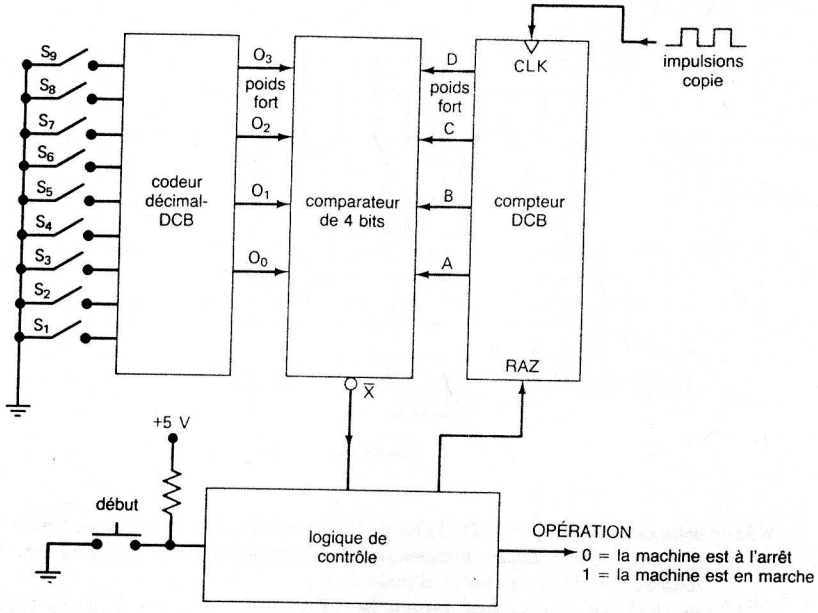


FIGURE 9-55

9-17 Le circuit de clavier de la figure 9-16 a été conçu pour accepter un nombre décimal de 3 chiffres. Que se produit-il si on appuie sur *quatre* touches numériques (par exemple 3095)? Concevez le circuit logique qu'il faut ajouter à ce circuit pour qu'après les trois premiers chiffres, tous les chiffres introduits ne soient pas pris en considération tant qu'on n'a pas appuyé sur la touche REMISE À ZÉRO. Autrement dit, si on introduit 3095 à partir du clavier, les registres de sortie afficheront 309 et ne tiendront pas compte du 5 et des nombres suivants à moins que le circuit ne soit initialisé (touche REMISE À ZÉRO).

9-18 Un technicien met à l'essai le circuit expérimental d'introduction de données au moyen d'un clavier de la figure 9-16 et vérifie son fonctionnement en introduisant des séries de nombres de trois chiffres. Il remarque que parfois le nombre « 0 » est affiché au lieu du chiffre qu'il a introduit. Il observe également que ceci se produit avec toutes les touches de façon plus ou moins aléatoire, quoi que ce soit plus fréquent dans le cas de certaines touches plutôt que d'autres. Il remplace tous les CI sans que cela règle le problème. Dites laquelle des pannes de circuit suivantes pourrait être la cause d'un tel fonctionnement. Justifiez votre choix.

a) Le technicien a oublié de mettre à la masse les entrées inutilisées de la porte OU.
b) Il a par inadvertance utilisé la borne $\overline{Q}$ au lieu de la borne Q du monostable.
c) Les rebonds des interrupteurs des touches durent plus de 20 ms.
d) Les sorties Y et Z sont en court-circuit l'une avec l'autre.

9-19 Même question que 9-18 mais cette fois-ci les registres et les afficheurs demeurent toujours à 0 quel que soit le nombre de fois qu'on appuie sur une touche.

9-20 Au cours de l'essai du circuit de la figure 9-16, un technicien s'aperçoit que chaque fois qu'une touche impaire est enfoncée, le chiffre affiché est correct, alors que chaque fois qu'il s'agit d'une touche paire, le chiffre est erroné. L'erreur observée est comme suit: l'enfoncement de « 0 » donne « 1 », de « 2 » donne « 3 », de « 4 » donne « 5 », etc. Considérez chacune des explications suivantes comme une cause possible de cette défaillance. Pour chacune, indiquez pourquoi c'est ou ce n'est pas la véritable cause du problème.

a) Une connexion ouverte de la sortie de l'inverseur POIDS FAIBLE aux entrées D des bascules.
b) La sortie D de la bascule Q_8 est court-circuitée par l'intérieur à V_{CC}.
c) Un pont de soudure met $\overline{O}_0$ en court-circuit avec la masse.

9-21 Un technicien soumet à un test le circuit de la figure 9-4 en suivant la procédure de l'exemple 9-7; ses résultats sont: toutes les sorties fonctionnent sauf celles de $\overline{O}_{16}$ à $\overline{O}_{19}$ et celles de $\overline{O}_{24}$ à $\overline{O}_{27}$ qui restent en permanence au niveau HAUT. Indiquez la panne de circuit la plus probable.

9-22 Un technicien teste le circuit de la figure 9-4 en suivant la procédure de l'exemple 9-7; il constate que la sortie fautive est activée par tous les codes d'entrée possibles, sauf ceux du tableau suivant. Examinez ce tableau et déterminez la cause probable de ce dysfonctionnement.

Code d'entrée $A_4A_3A_2A_1A_0$	Sorties activées
10000	$\overline{O}_{16}$ et $\overline{O}_{24}$
10001	$\overline{O}_{17}$ et $\overline{O}_{25}$
10010	$\overline{O}_{18}$ et $\overline{O}_{26}$
10011	$\overline{O}_{19}$ et $\overline{O}_{27}$
10100	$\overline{O}_{20}$ et $\overline{O}_{28}$
10101	$\overline{O}_{21}$ et $\overline{O}_{29}$
10110	$\overline{O}_{22}$ et $\overline{O}_{30}$
10111	$\overline{O}_{23}$ et $\overline{O}_{31}$

9-23 Supposons qu'une résistance de 22 Ω se retrouve par erreur à la place du segment *g* de la figure 9-8. Comment cette résistance modifie-t-elle l'affichage? Quels problèmes pourront survenir?

9-24 Même problème que celui de l'exemple 9-8, mais cette fois-ci pour la séquence suivante:

Nombre	0	1	2	3	4	5	6	7	8	9
Affichage	0	1	2	3	8	9	c	ɔ	4	5

9-25 Même problème que celui de l'exemple 9-8, mais cette fois-ci pour la séquence suivante:

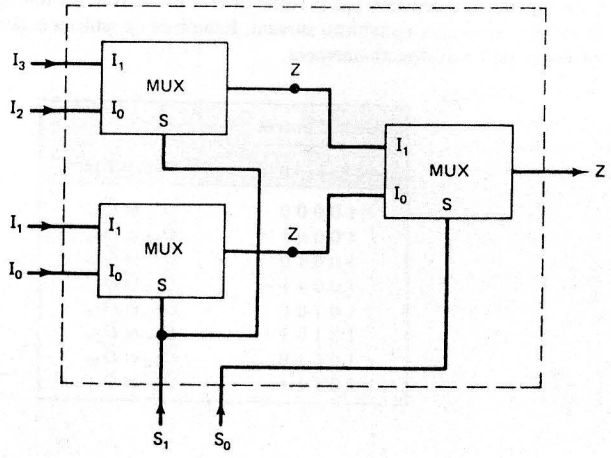
Nombre	0	1	2	3	4	5	6	7	8	9
Affichage	0	7	2	3	9	9	8	7	8	9

9-26 Pour mettre à l'essai le circuit de la figure 9-11, une technicienne a raccordé un compteur DCB aux entrées 4511b et elle envoie des impulsions au compteur à un rythme très lent. Elle note que le segment *f* fonctionne de façon instable et cela, sans raison évidente. Nommez quelques causes possibles d'un tel dysfonctionnement. (*Suggestion:* rappelez-vous que ce CI est un CMOS.)

SECTIONS 9-7 et 9-8

9-27 Le circuit de la figure 9-56 comprend 3 multiplexeurs à deux entrées (figure 9-20). Trouvez la fonction réalisée par ce circuit.

FIGURE 9-56



9-28 Exploitez le principe illustré par le dernier problème pour agencer plusieurs multiplexeurs 74151 1 parmi 8 afin d'obtenir un multiplexeur 1 parmi 64.

9-29 Montrez comment deux 74157 et un 74151 peuvent être réunis pour former un multiplexeur 1 parmi 16 sans devoir utiliser d'autres portes logiques. Désignez les entrées I_0–I_{15} afin de montrer la correspondance avec le code de sélection.

9-30 a) Améliorez le circuit de la figure 9-25 pour pouvoir afficher le contenu de deux compteurs DCB à trois étages.

b) Calculez le nombre de connexions dans ce circuit et comparez-le avec le nombre requis s'il y avait pour chaque étage de chaque compteur un pilote/décodeur et un afficheur séparés.

9-31 La figure 9-57 nous présente une façon d'utiliser un multiplexeur pour générer des formes d'ondes logiques ayant l'aspect que l'on veut. On programme la forme de l'onde au moyen des 8 interrupteurs unipolaires, deux directions; la forme d'onde est produite indéfiniment tant que des impulsions arrivent au compteur MODULO-8. Tracez la forme d'onde que l'on observe en *Z* pour les positions des interrupteurs indiquées.

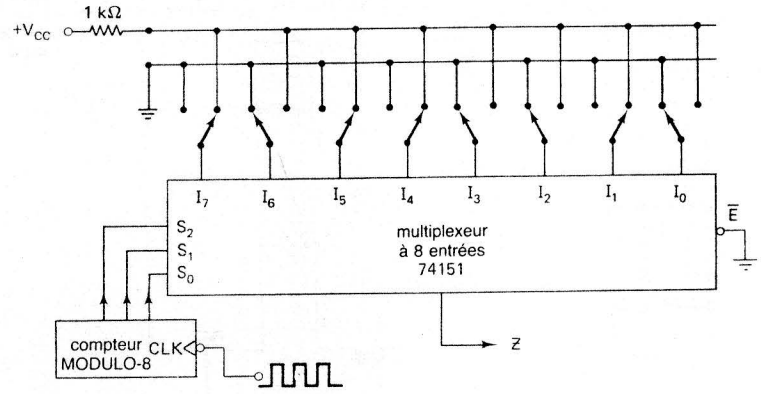


FIGURE 9-57

9-32 Changez le compteur MODULO-8 de la figure 9-57 pour un compteur MODULO-16 et raccordez la ligne correspondant au bit de poids le plus fort à l'entrée $\overline{E}$ du multiplexeur. Tracez la forme d'onde de *Z*.

9-33 Montrez comment on peut utiliser un 74151 pour générer la fonction logique: $Z = AB + BC + AC$.

9-34 Montrez comment un multiplexeur à 16 entrées peut être employé pour générer la fonction $Z = \overline{A}B\overline{C}D + BCD + A\overline{B}\overline{D} + AB\overline{C}D$.

9-35 Le circuit de la figure 9-58 représente un MULTIPLEXEUR à 8 entrées que l'on peut utiliser pour réaliser une fonction logique à quatre variables, et cela même si le multiplexeur ne dispose que de trois entrées de sélection. Trois des variables, soit *A*, *B* et *C* sont raccordées aux entrées de sélection. La quatrième, *D*, et son complément $\overline{D}$ sont connectés aux entrées de données du multiplexeur, nécessaires à la réalisation de la fonction logique. Les autres entrées du boîtier sont forcées à la valeur BASSE ou HAUTE selon les nécessités de la fonction.

a) Dressez la table de vérité de la sortie Z pour les 16 combinaisons d'entrée possibles.
b) Mettez sous forme d'une somme de produits l'expression de Z et simplifiez-la; vérifiez que:

$$Z = \overline{C}B\overline{A} + D\overline{C}B\overline{A} + \overline{D}C\overline{B}\overline{A}$$

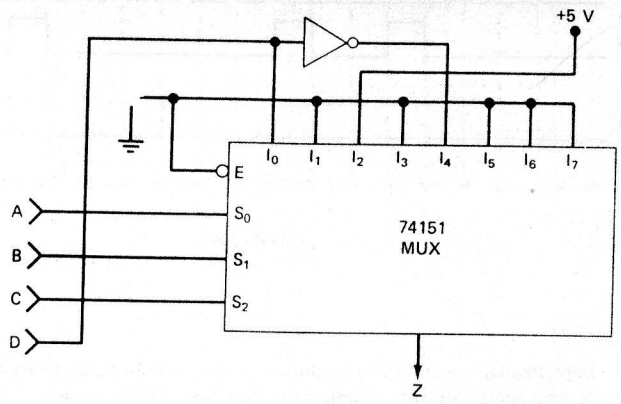


FIGURE 9-58

9-36 La méthode mise en pratique à la figure 9-58 peut être appliquée à toute fonction logique à quatre variables; voici comment procéder:

1. Dressez la table de vérité de la fonction désirée dont la sortie est Z.
2. Écrivez la somme de produits pour Z; ne la simpfiez pas. Par exemple:

$$Z = DC\overline{B}A + \overline{D}CB\overline{A} + DCB\overline{A} + D\overline{C}BA + \overline{D}\overline{C}BA + \overline{D}C\overline{B}A$$

3. Recherchez les combinaisons identiques de C, B et A, puis mettez en facteurs:

$$\begin{aligned} Z &= DC\overline{B}A + CB\overline{A}(\overline{D} + D) + \overline{C}BA(\overline{D} + D) + \overline{D}\overline{C}BA \\ &= DC\overline{B}A + CB\overline{A} + \overline{C}BA + \overline{D}\overline{C}BA \end{aligned}$$

4. Ne considérez que les termes contenant C, B et A aussi bien dans leur forme normale que dans leur forme complémentée. Pour chacun d'eux, connectez l'entrée de données correspondante du multiplexeur à un niveau HAUT:

$CB\overline{A}$ → connexion d'un niveau HAUT à l'entrée I_6
$\overline{C}BA$ → connexion d'un niveau HAUT à l'entrée I_3

5. Ne considérez que les termes renfermant la variable D. Connectez la variable D ou $\overline{D}$ à l'entrée du multiplexeur qui correspond aux variables CBA:

$DC\overline{B}A$ → connexion de D à l'entrée I_5
$\overline{D}\overline{C}BA$ → connexion de $\overline{D}$ à l'entrée I_1

6. Connectez les entrées restantes du multiplexeur à un niveau BAS.
   a) Vérifiez le branchement de la figure 9-58 au moyen de cette méthode.
   b) Mettez cette méthode en pratique pour réaliser une fonction qui donne un niveau HAUT seulement quand les quatre variables d'entrée sont au même niveau, ou bien quand les variables B ou C sont à des niveaux opposés.

SECTION 9-9

9-37 CONTRÔLE
Pour chacun des énoncés suivants, dites s'il caractérise un décodeur, un codeur, un multiplexeur ou un démultiplexeur:
a) Possède plus d'entrées que de sorties.
b) Utilise des entrées de SÉLECTION.
c) Peut servir à réaliser une conversion parallèle-série.
d) Fournit en sortie un code binaire.
e) Une seule de ses sorties est active à la fois.
f) On peut s'en servir pour diriger un signal d'entrée vers une sortie parmi plusieurs.
g) Peut servir à réaliser des fonctions logiques.

9-38 Indiquez comment on peut utiliser un décodeur 7442 pour qu'il agisse comme un démultiplexeur 1 parmi 8. (*Suggestion:* voir problème 9-7.)

9-39 Appliquez comme ci-dessous les formes d'ondes de la figure 9-54 aux entrées du démultiplexeur 74LS138 de la figure 9-31 a):

$$D \rightarrow A_2 \qquad C \rightarrow A_1 \qquad B \rightarrow A_0 \qquad A \rightarrow E_1$$

Dessinez les formes d'ondes sur les sorties du démultiplexeur.

9-40 Soit le système de la figure 9-33. Supposez que la fréquence de l'horloge est 10 ips. Décrivez l'état de la console de télésurveillance pour chacun des cas suivants.
a) Toutes les portes sont fermées.
b) Toutes les portes sont ouvertes.
c) Les portes 2 et 6 sont ouvertes.

9-41 Modifiez le système de la figure 9-33 pour qu'il contrôle 16 portes. Utilisez un multiplexeur à 16 entrées 74150 et deux démultiplexeurs 74LS138. Combien de conducteurs aboutissent à la console de télésurveillance.

9-42 Tracez les formes d'ondes de Z, O_0, O_1, O_2 et O_3 de la figure 9-32 quand les données contenues dans les registres sont respectivement: $[A] = 0011$, $[B] = 0110$, $[C] = 1001$, $[D] = 0111$.

SECTION 9-11

9-43 Soit l'ordonnanceur de commande de la figure 9-27. Décrivez l'effet de chacune des pannes suivantes sur son fonctionnement.
a) L'entrée I_3 du multiplexeur est court-circuitée à la masse.
b) Les connexions entre les capteurs 3 et 4 et le multiplexeur sont interverties.

9-44 Soit le circuit de la figure 9-25. Un essai de ce circuit a donné les résultats suivants:

		Nombre réel	Nombre affiché
Cas 1	Compteur 1	33	33
	Compteur 2	47	47
Cas 2	Compteur 1	82	02
	Compteur 2	64	64
Cas 3	Compteur 1	63	63
	Compteur 2	95	15

Indiquez les causes probables de ce mauvais fonctionnement.

9-45 Un test du système de surveillance de la sécurité des lieux donne les résultats suivants:

Condition	DEL
Toutes les portes fermées	Toutes éteintes
Porte 0 ouverte	DEL 0 clignote
Porte 1 ouverte	DEL 2 clignote
Porte 2 ouverte	DEL 1 clignote
Porte 3 ouverte	DEL 3 clignote
Porte 4 ouverte	DEL 4 clignote
Porte 5 ouverte	DEL 6 clignote
Porte 6 ouverte	DEL 5 clignote
Porte 7 ouverte	DEL 7 clignote

Quelles sont les causes possibles de ce comportement?

9-46 Un test du système de surveillance de la sécurité des lieux donne les résultats suivants:

Condition	DEL
Toutes les portes fermées	Toutes éteintes
Porte 0 ouverte	DEL 0 clignote
Porte 1 ouverte	DEL 1 clignote
Porte 2 ouverte	DEL 2 clignote
Porte 3 ouverte	DEL 3 clignote
Porte 4 ouverte	DEL 4 clignote
Porte 5 ouverte	DEL 5 clignote
Porte 6 ouverte	Aucune ne clignote
Porte 7 ouverte	Aucune ne clignote
Portes 6 et 7 ouvertes	DEL 6 et 7 clignotent

Quelles sont les causes possibles de ce comportement?

9-47 Supposez que le système de transmission de données synchrone de la figure 9-32 ne fonctionne pas bien et que l'on observe ceci: la forme d'onde de Z est exacte, mais la forme d'onde de O_0 est identique en tout temps à la forme d'onde de Z tandis que les autres sorties sont toujours au niveau BAS. Dites quelle panne explique ce mauvais fonctionnement. Supposez que tous les circuits sont des TTL, justifiez votre choix.

9-48 Le système de transmission synchrone de la figure 9-34 manifeste un fonctionnement défaillant; sur un oscilloscope haute fréquence, on observe les formes d'ondes reproduites à la figure 9-59. Remarquez les pointes dans le signal O_1. Envisagez les deux défaillances suivantes. Pour chacune, indiquez si elle peut être ou non la cause du problème.

a) Les connexions aux broches S_1 et S_0 du démultiplexeur sont interverties.

b) Les connexions aux broches Q_1 et Q_0 du compteur de mot du récepteur sont interverties.

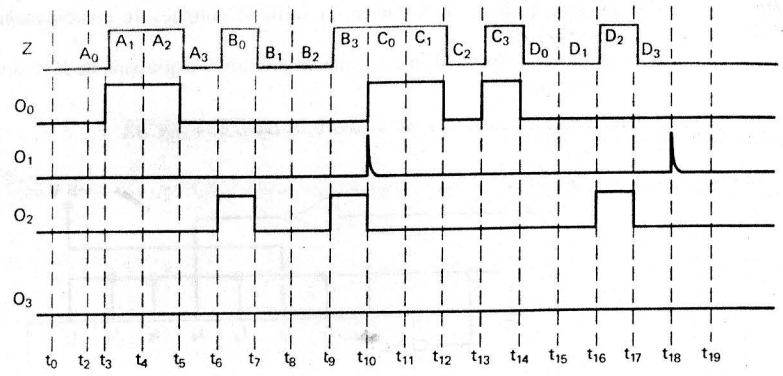


FIGURE 9-59

SECTION 9-12

9-49 Reprenez la conception du circuit du problème 9-16 en employant un comparateur de grandeurs 74LS85. Ajoutez une fonction « dépassement d'exemplaires » qui arme une sortie ALARME si jamais la sortie OPÉRATION fait défaut d'arrêter la machine quand le nombre d'exemplaires requis est atteint.

9-50 Montrez comment câbler des 74LS85 pour qu'ils comparent des nombres de 10 bits.

SECTION 9-13

9-51 Supposez une valeur DCB de 69 en entrée du convertisseur de code de la figure 9-43. Déterminez les niveaux sur chaque sortie Σ et la sortie binaire finale.

9-52 Un technicien met à l'essai le convertisseur de code de la figure 9-43 et relève les observations suivantes:

Entrée DCB	*Sortie binaire*
52	0110101
99	1100100
27	0011011

Quelle est la cause probable de cette défectuosité?

SECTIONS 9-14 et 9-15

9-53 Pour le montage de bus de la figure 9-46, décrivez les exigences du signal d'entrée qui assurent le transfert simultané du contenu du registre C dans les deux autres registres.

9-54 Supposez que les registres de la figure 9-46 sont initialement à $[A] = 1011$, $[B] = 1000$ et $[C] = 0111$. On applique les signaux illustrés à la figure 9-60 aux entrées du registre.

a) Indiquez quel est le contenu de chacun des registres aux instants t_1, t_2 et t_3.

b) Que se produit-il si $\overline{IE}_A$ est BAS à l'arrivée de la troisième impulsion.

9-55 Pour les mêmes conditions initiales qu'au problème 9-54, tracez le signal sur DB_3 pour les formes d'ondes de la figure 9-60.

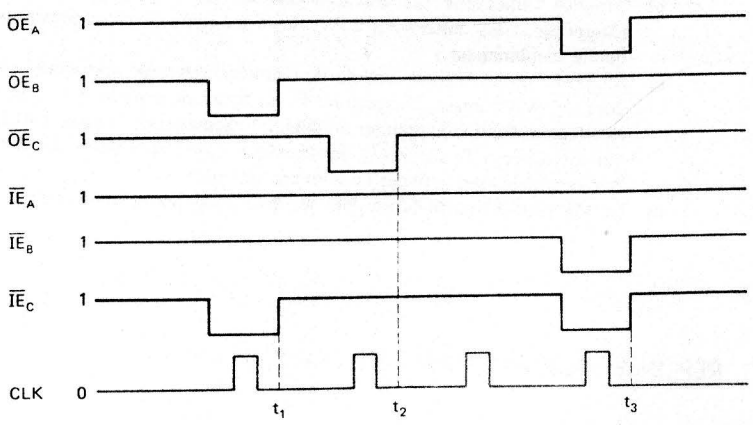


FIGURE 9-60

9-56 La figure 9-61 montre deux dispositifs que l'on veut brancher sur le bus de données de la figure 9-46. Un de ceux-ci est un ensemble d'interrupteurs avec tampon qui nous permet d'introduire manuellement des données dans n'importe quel registre du bus. L'autre dispositif est un registre de sortie qui nous permet de verrouiller n'importe quelle donnée qui se trouve sur le bus durant un transfert de données et de la communiquer à des DEL pour qu'elles l'affichent.

a) Supposez que tous les registres sont à 0000. Donnez les grandes lignes de la séquence des opérations nécessaires pour charger dans les registres, au moyen des interrupteurs, les données que voici: $[A] = 1011$, $[B] = 0001$, $[C] = 1110$.

b) Indiquez quel état les DEL afficheront à la fin de la séquence.

9-57 Maintenant qu'on a raccordé au bus de la figure 9-46 les dispositifs de la figure 9-61, il y a 5 dispositifs sur ce bus. Le circuit de la figure 9-62 a) sert à produire des signaux de validation qui conduisent les différents transferts effectués par l'entremise du bus de données. On retrouve dans ce circuit un boîtier 74LS139 qui renferme deux décodeurs 1 parmi 4 indépendants ayant une borne de validation vraie au niveau BAS. Le décodeur supérieur choisit le dispositif qui place des données sur le bus (sélection de la sortie) et le décodeur inférieur choisit le dispositif qui reçoit les données du bus (sélection de l'entrée). Supposez que les sorties de ce circuit soient connectées aux entrées validation correspondantes des dispositifs branchés sur le bus. Supposez également que tous les registres sont initialement à 0000 au temps t_0 et que les interrupteurs occupent les positions indiquées sur la figure 9-61.

a) Déterminez le contenu de chaque registre aux instants t_1, t_2 et t_3 en réponse aux formes d'ondes de la figure 9-62 b).

b) Dites s'il y a pour ce circuit « conflit d'utilisation du bus ».

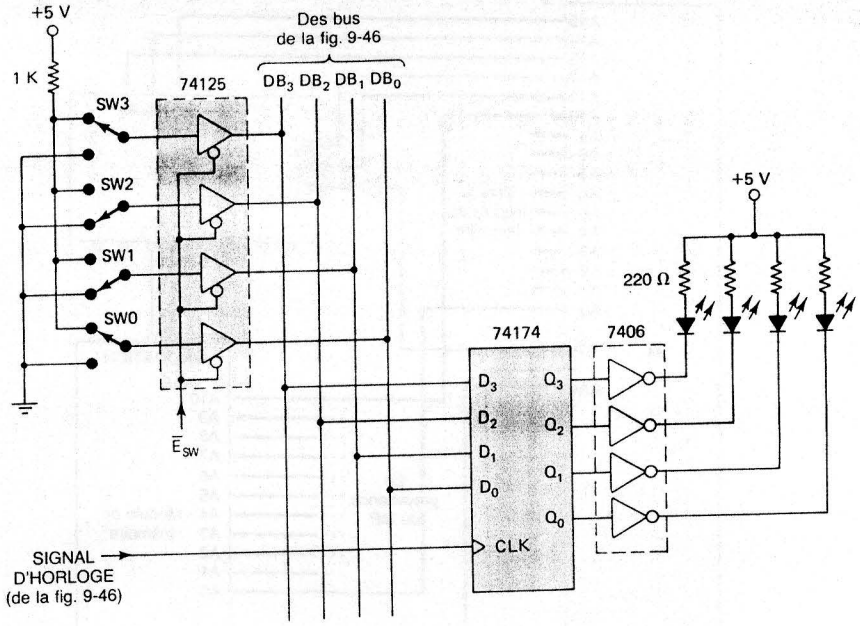


FIGURE 9-61

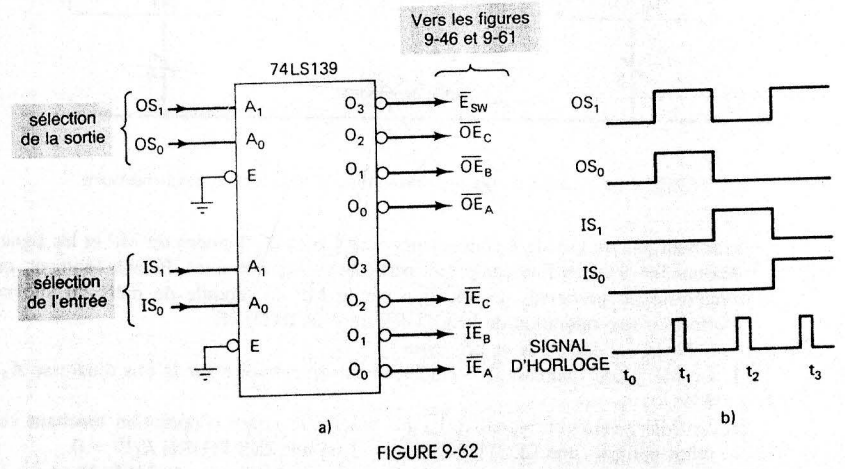


FIGURE 9-62

APPLICATIONS AU MICRO-ORDINATEUR

9-58 La figure 9-63 nous montre les circuits de base d'une interface entre le micro-processeur (MP) et un module de mémoire. Ce module de mémoire contient un ou plusieurs boîtiers de CI en mesure de recevoir des données du MP (fonction ÉCRITURE), ou d'en envoyer au MP (fonction LECTURE). Les données ainsi échangées

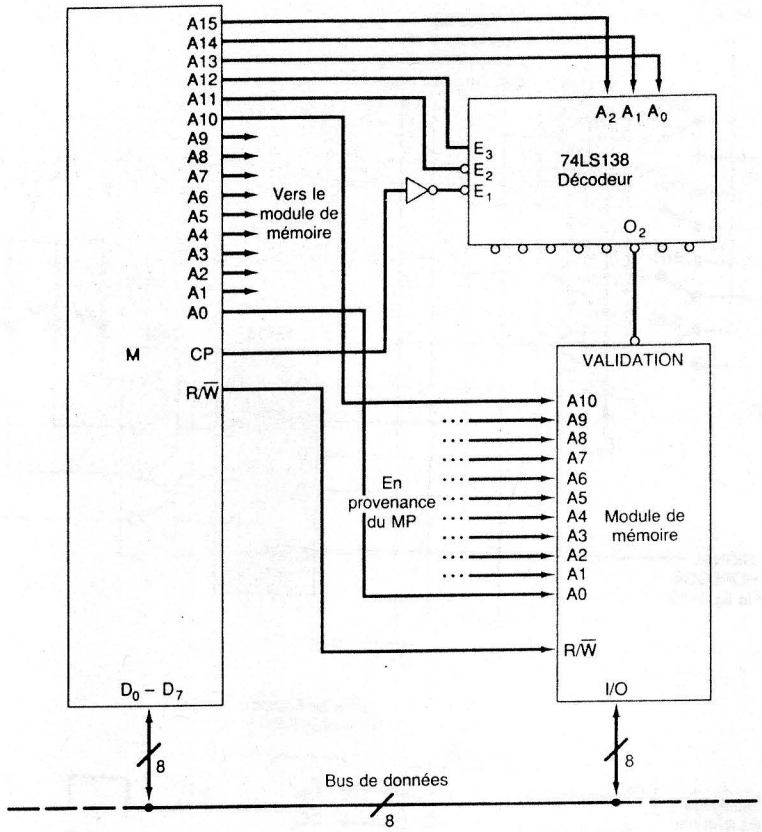


FIGURE 9-63 Circuit de base d'une interface microprocesseur-mémoire.

transitent par un bus de 8 conducteurs. Les lignes de données du MP et les lignes de données des E/S font également partie de ce bus commun. Pour le moment, on s'intéresse au processus de sélection par le MP du module de mémoire en vue d'effectuer une opération de LECTURE ou d'ÉCRITURE.

Voici les étapes de ce processus:

1. Le MP place l'adresse de l'emplacement de mémoire sur le bus d'adresse A_{15} à A_0.
2. Le MP génère un signal $R/\overline{W}$ qui précise le genre d'opération touchant cet emplacement: une LECTURE $R/\overline{W} = 1$ ou une ÉCRITURE $R/\overline{W} = 0$.
3. Les cinq premiers bits du bus d'adresse sont décodés par le 74LS138, qui a la maîtrise de l'entrée VALIDATION du module de la mémoire. Cette entrée VALIDATION doit être active pour que le module de mémoire puisse être LU ou « ÉCRIT ».
4. Les 11 autres bits d'adresse sont connectés au module de mémoire qui les utilise si le signal VALIDATION est actif pour choisir l'emplacement bien précis dans la mémoire *interne* auquel demande d'accéder le MP.

Pour que le module de mémoire puisse être lu ou écrit, il faut que le microprocesseur place la bonne adresse sur le bus d'adresse afin de valider cette mémoire, puis il doit faire passer *CP* momentanément au niveau HAUT.

a) S'il y a lieu, déterminez parmi les adresses hexadécimales suivantes celle qui valide le module de mémoire: 697F; 57FA; 5F00.

b) Trouvez l'intervalle des adresses hexadécimales qui déclenche la mémoire (*Suggestion:* les entrées A_0 à A_{10} de la mémoire peuvent prendre n'importe quelle combinaison.)

c) Supposez qu'un second module de mémoire identique soit ajouté à ce circuit dont le bus d'adresse, le signal $R/\overline{W}$, les lignes de données I/O sont connectés exactement comme le premier module, à l'exception de l'entrée VALIDATION qui est reliée à la sortie $\overline{O}_4$ du décodeur. Quel est l'intervalle des adresses hexadécimales qui activent ce nouveau module?

d) Le MP peut-il lire ou écrire dans les deux modules en même temps?

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 9-1

1. Non. **2.** L'entrée de validation autorise ou non les circuits logiques du décodeur à reconnaître ou non le code d'entrée binaire. **3.** Le 7445 possède des sorties à collecteur ouvert qui peuvent soutenir jusqu'à 30 V et 80 mA. **4.** 24 broches, 2 validations, 16 entrées, 4 sorties, V_{CC}, et la masse.

SECTION 9-2

1. *a, b, c, f, g* **2.** Vrai.

SECTION 9-3

1. DEL: a, e, f; ACL: b, c, d, e

SECTION 9-4

1. Un codeur produit un code de sortie correspondant à l'entrée activée. Un décodeur valide une sortie correspondant au code binaire appliqué. **2.** Dans un codeur à priorité, le code de sortie correspond à l'entrée activée ayant le chiffre le *plus élevé*. **3.** DCB normal = 0110. **4. a)** Génère un FM quand une touche est enfoncée. **b)** Convertit l'enfoncement de la touche en son code DCB. **c)** Produit une impulsion sans rebondissement en vue de commander le compteur à anneau. **d)** Constitue un compteur à anneau qui déclenche selon une séquence les registres de sortie. **e)** Mémorise les codes DCB générés par les enfoncements des touches.

SECTION 9-5

1. Décodeur DCB-décimal. **2.** Priorité élevée. **3.** Tampon ou pilote.

SECTION 9-7

1. Le nombre binaire positionné sur les entrées de sélection détermine l'entrée de donnée qui franchit le multiplexeur jusqu'à la sortie. **2.** Trente-deux entrées de donnée et cinq entrées de sélection.

SECTION 9-8

1. La conversion parallèle-série, l'acheminement des données, la réalisation de fonction logique, l'ordonnancement d'opérations. **2.** Faux — elles sont appliquées aux entrées de sélection. **3.** Compteur.

SECTION 9-9

1. Un multiplexeur choisit une entrée parmi plusieurs dont les signaux arrivent jusqu'à la sortie; un démultiplexeur choisit une sortie parmi plusieurs qui reçoit le signal d'entrée. **2.** Vrai, à la condition que le décodeur possède une entrée de VALIDATION. **3.** Les DEL s'allument et s'éteignent conformément à une séquence.

SECTION 9-12

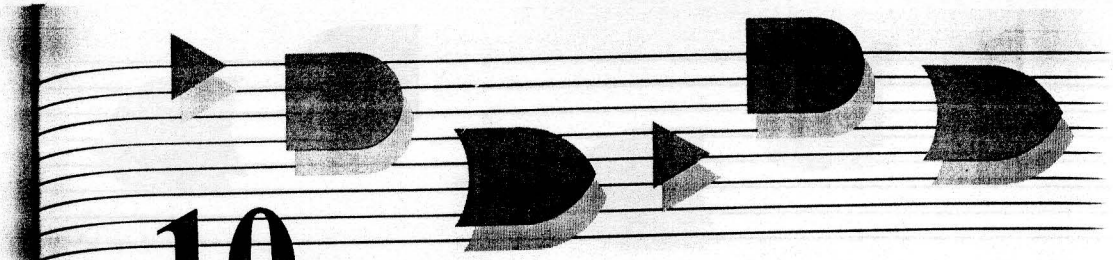
1. Pour étendre les opérations de comparaison à des nombres de plus de 4 bits. **2.** $O_{A=B} = 1$; les autres sorties sont à 0.

SECTION 9-13

1. Un convertisseur de code prend la donnée d'entrée exprimée dans un code binaire et trouve son équivalent dans un autre code binaire. **2.** Ces chiffres représentent les valeurs décimales jusqu'à 999. Pour représenter 999 dans le code binaire pur, il faut 10 bits.

SECTION 9-15

1. Un ensemble de conducteurs de liaison sur lesquels sont connectées les entrées et les sorties de différents dispositifs. **2.** Un conflit de bus survient quand les sorties de plus d'un dispositif essaient de s'approprier au même instant l'usage du bus. On prévient cette situation au moyen du contrôle d'entrées de validation des dispositifs. **3.** Pour obtenir une plus petite résistance de sortie et parvenir à piloter les nombreuses entrées connectées au bus. **4.** Réduire le nombre de broches du boîtier du registre et le nombre de connexions sur le bus de données. **5.** Voyez figure 9-51.

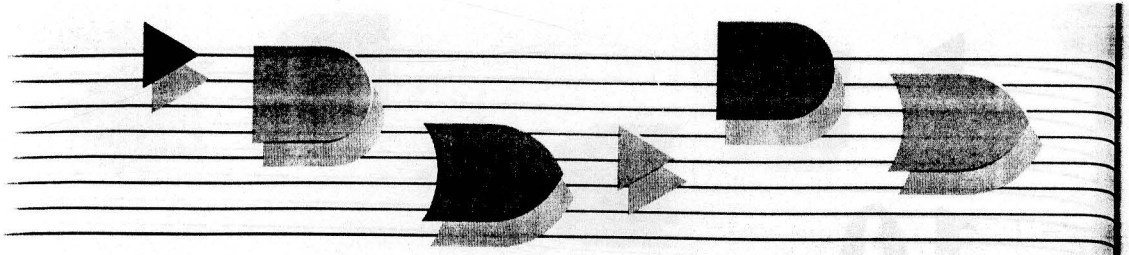


10

CONVERTISSEURS ANALOGIQUES-NUMÉRIQUES ET NUMÉRIQUES-ANALOGIQUES

Contenu

10-1 Les échanges avec un monde analogique
10-2 Conversion numérique-analogique
10-3 Circuits des convertisseurs NA
10-4 Spécifications des CNA
10-5 Applications des CNA
10-6 Dépannage des CNA
10-7 Conversion analogique-numérique
10-8 CAN rampe numérique
10-9 Acquisition des données
10-10 CAN par approximations successives
10-11 CAN parallèles
10-12 Autres méthodes de conversion analogique-numérique
10-13 Voltmètre numérique
10-14 Circuit échantillonnage et mémorisation
10-15 Multiplexage
19-16 Oscilloscope numérique à mémoire



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Comprendre la théorie de fonctionnement de plusieurs types de convertisseurs numériques-analogiques et les limites inhérentes de ces circuits.
- Lire et interpréter les diverses spécifications des fabricants de CAN.
- Mettre en oeuvre différentes procédures d'essai visant à dépanner les circuits CAN.
- Comparer les avantages et les inconvénients respectifs du CAN rampe numérique, du CAN par approximations successives et du CAN parallèle.
- Analyser le processus de numérisation d'un signal analogique effectué par un ordinateur et un CAN et de recomposition d'un signal analogique à partir de données numériques.
- Décrire le principe de fonctionnement d'un voltmètre numérique.
- Comprendre la nécessité d'intercaler des circuits échantillonnage et mémorisation avant les CAN.
- Énumérer les avantages et les limitations du multiplexage des CAN.

10-1 LES ÉCHANGES AVEC UN MONDE ANALOGIQUE

Révision des grandeurs numériques et analogiques Une grandeur *numérique* est une variable dont le domaine des valeurs possibles est 0 et 1, ou BAS et HAUT, ou vrai et faux. En pratique, une grandeur numérique, comme une tension, peut correspondre à une valeur quelconque à l'intérieur d'une plage préétablie; par définition, toutes les valeurs dans une plage possèdent la même valeur numérique. C'est ainsi qu'en logique TTL, nous savons que:

0 V à 0,8 V = 0 logique
2 V à 5 V = 1 logique

Toute tension qui se situe dans la plage 0 V et 0,8 V est considérée comme la valeur numérique 0, alors que toute tension entre 2 V et 5 V est considérée comme la valeur numérique 1. La tension exacte n'a pas d'importance, puisque les circuits numériques réagissent de la même façon à toutes les valeurs de tension se situant dans la plage précisée.

À l'inverse, une grandeur *analogique* peut prendre n'importe quelle valeur d'une plage continue et, ce qui est le plus important, chacune de ces valeurs est significative. Par exemple, on mesure 2,76 V à la sortie d'un convertisseur température-tension, ce qui

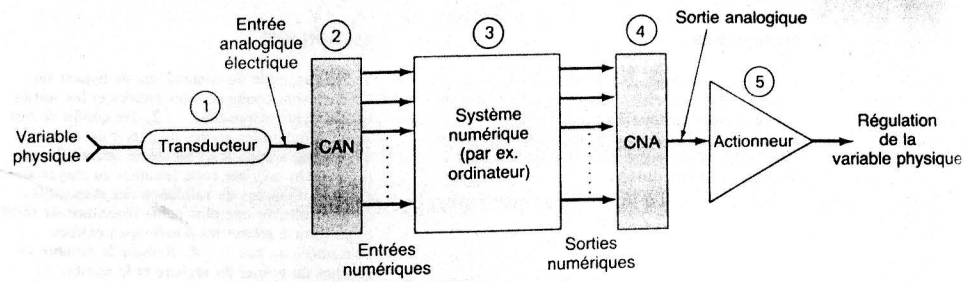


FIGURE 10-1 Convertisseur analogique-numérique (CAN) et convertisseur numérique-analogique (CNA) servant d'interfaces à un ordinateur de régulation pour qu'il puisse échanger avec le monde analogique extérieur.

correspond à une température de 27,6 °C. Si la tension mesurée est différente, 2,34 V ou 3,78 V, la température correspondante est tout à fait différente. Autrement dit, chaque valeur d'une grandeur analogique a une signification précise. Un autre exemple de grandeur analogique est la tension de sortie d'un amplificateur audio qui excite un haut-parleur. Cette tension est une grandeur analogique, car chaque valeur possible excite de façon distincte le haut-parleur.

La grande majorité des variables physiques sont de nature analogique et peuvent épouser n'importe quelle valeur à l'intérieur d'une plage continue. Parmi les grandeurs analogiques connues, mentionnons la température, la pression, l'intensité lumineuse, les signaux audio, le déplacement, la vitesse de rotation et le débit. Les systèmes numériques, par ailleurs, ne sont constitués que de circuits numériques et ne réalisent que des opérations numériques. Tous les renseignements que l'on fournit en entrée à un système numérique doivent se trouver sous forme numérique. Également, toutes les données que restituent le système numérique se trouvent sous cette forme. Un ordinateur de surveillance ou de régulation de procédé doit donc composer avec l'incompatibilité entre sa façon de traiter les données et le caractère analogique des grandeurs à régler. La figure 10-1 illustre cette situation. Ce schéma présente les cinq éléments que l'on retrouve toujours dans un montage de régulation industrielle par ordinateur, quand la variable physique est analogique:

1. *Transducteur.* Normalement, la variable physique est une grandeur non électrique. Un **transducteur** est un dispositif qui fournit un équivalent électrique de la variable physique. Voici quelques exemples de transducteurs: thermistors, cellules photoélectriques, photodiodes, débitmètres, transducteurs manométriques et tachymètres. La sortie électrique d'un transducteur est une tension ou un courant analogique proportionnel à la variable physique surveillée.

   Par exemple, la variable physique pourrait être la température de l'eau dans un grand réservoir alimenté par des conduites d'eau chaude et d'eau froide. Supposons que cette température varie entre 80 ° et 150 °F et qu'un thermistor, ainsi que ses circuits associés, convertissent cette température en une tension comprise entre 800 mV et 1500 mV. Remarquez que la sortie du transducteur est directement proportionnelle à la température: à chaque 1 °F correspond une tension de sortie de 10 mV. Ce facteur de proportionnalité a été choisi par souci de commodité.
2. *Convertisseur analogique-numérique (CAN).* La sortie électrique du transducteur est délivrée à l'entrée analogique du CAN. Ce dernier convertit la valeur

analogique en une grandeur numérique, c'est-à-dire en un certain nombre de bits qui représentent la grandeur analogique.

Poursuivons notre exemple: le CAN convertit les tensions comprises entre 800 et 1500 mV en des valeurs binaires se trouvant dans la plage 01010000 (80) et 10010110 (150). Comme la sortie binaire du CAN est proportionnelle à la tension d'entrée analogique, chaque unité de la sortie numérique vaut 10 mV.

3. *Ordinateur.* La représentation numérique de la variable réglée est transmise du CAN à l'ordinateur qui la mémorise et la traite conformément aux instructions du programme de régulation qu'il exécute. Ce programme peut effectuer des calculs ou d'autres traitements qui utilisent cette donnée; une fois établie, la valeur résultante sert à la régulation de la température.
4. *Convertisseur numérique-analogique (CNA).* La sortie numérique fournie par l'ordinateur arrive dans un CNA, dont le rôle est de faire correspondre à l'entrée numérique une sortie analogique proportionnelle. Par exemple, l'ordinateur délivre un nombre entre 00000000 et 11111111 et le CNA fournit en sortie une tension entre 0 et 10 V.
5. *Actionneur.* Le signal analogique que l'on obtient du CNA est connecté à un circuit ou à un dispositif qui agit comme actionneur afin de régler effectivement la variable physique contrôlée. Toujours dans notre exemple de la température de l'eau, l'actionneur pourra être une soupape électrique qui contrôle le débit de l'eau chaude arrivant dans le réservoir selon la valeur de la tension analogique du CNA. Le débit pourra être proportionnel à cette tension: à 0 V correspond un débit nul, et à 10 V, un débit maximal.

On voit donc que le CAN et le CNA jouent le rôle d'*interfaces* entre un système entièrement numérique et le monde analogique. Ce rôle est de plus en plus essentiel depuis l'arrivée de micro-ordinateurs bon marché dans des secteurs de la régulation industrielle où la régulation informatisée était précédemment impossible à réaliser.

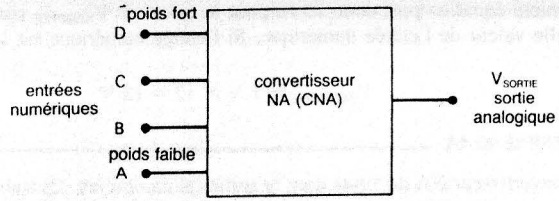
QUESTIONS DE RÉVISION

1. Quel est le rôle d'un transducteur?
2. Quel est le rôle d'un CAN?
3. Que fait le plus souvent un ordinateur avec les données qu'il reçoit d'un CAN?
4. Quel est le rôle d'un CNA?
5. Quel est le rôle d'un actionneur?

10-2 CONVERSION NUMÉRIQUE-ANALOGIQUE

Les deux opérations d'entrée et de sortie les plus importantes dans un contexte de régulation de procédé sont la conversion numérique-analogique et la conversion analogique-numérique. Étant donné que de nombreuses méthodes de conversion AN reposent sur des méthodes utilisées pour la conversion NA, nous étudierons en premier lieu la conversion NA.

La *conversion NA* se définit comme suit: transposition d'une valeur déjà exprimée selon une représentation *numérique* (comme en binaire pur ou en DCB) en une tension ou un courant proportionnel à cette valeur numérique. La figure 10-2 a) nous fait voir le



a)

D	C	B	A	V_{SORTIE}	
0	0	0	0	0	volts
0	0	0	1	1	
0	0	1	0	2	
0	0	1	1	3	
0	1	0	0	4	
0	1	0	1	5	
0	1	1	0	6	
0	1	1	1	7	
1	0	0	0	8	
1	0	0	1	9	
1	0	1	0	10	
1	0	1	1	11	
1	1	0	0	12	
1	1	0	1	13	
1	1	1	0	14	
1	1	1	1	15	volts

b)

FIGURE 10-2 Convertisseur NA de 4 bits dont la sortie est une tension.

schéma fonctionnel d'un convertisseur NA de 4 bits type. Pour l'instant, laissons de côté les circuits internes du convertisseur et examinons les divers rapports entre les entrées et les sorties.

Les entrées numériques D, C, B et A proviennent généralement du registre de sortie du système numérique. Les $2^4 = 16$ nombres binaires différents qu'on peut représenter avec ces 4 bits sont énumérés à la figure 10-2 b). Pour chaque nombre d'entrée, la tension de sortie du convertisseur NA est différente. En fait, dans l'exemple actuel, la tension de sortie analogique V_{SORTIE} est une grandeur dont l'amplitude, exprimée en volts, est égale au nombre binaire. Il existe des convertisseurs dont la tension de sortie est égale à deux fois le nombre binaire, ou au nombre binaire multiplié par un facteur de proportionnalité. Le même principe est aussi vrai si la sortie du CNA est un courant I_{SORTIE}.

En général,

$$\textbf{sortie analogique} = K \times \textbf{entrée numérique} \qquad (10\text{-}1)$$

où K est le facteur de proportionnalité, une constante pour un certain CNA. La sortie analogique est évidemment une tension ou un courant. S'il s'agit d'une tension, K a les unités d'une tension, s'il s'agit d'un courant, ce facteur a les unités d'un courant. Pour le cas du CNA de la figure 10-2, $K = 1$ V, donc:

$$V_{SORTIE} = (1 \text{ V}) \times \text{entrée numérique}$$

Cette dernière équation peut donc servir pour le calcul de V_{SORTIE} correspondant à n'importe quelle valeur de l'entrée numérique. Si l'entrée numérique est $1100_2 = 12_{10}$, nous obtenons:

$$V_{SORTIE} = 1 \text{ V} \times 12 = 12 \text{ V}$$

EXEMPLE 10-1A

Soit un convertisseur NA de 5 bits dont la sortie est un courant. Quand l'entrée numérique est 10100, le courant de sortie produit est 10 mA. Calculez l'intensité de I_{SORTIE} pour une entrée numérique de 11101.

Solution L'entrée numérique 10100 est l'équivalent binaire de 20_{10}. Puisque le courant correspondant à cette entrée est $I_{SORTIE} = 10$ mA, on déduit un facteur de proportionnalité de 0,5; c'est donc dire que $I_{SORTIE} = 0{,}5 \times$ valeur binaire. Donc le courant correspondant à l'entrée binaire 11101, équivalent à 29_{10}, est

$$I_{SORTIE} = 0{,}5 \times 29$$
$$= 14{,}5 \text{ mA}$$

Le facteur de proportionnalité, K, varie d'un CNA à l'autre.

EXEMPLE 10-1B

Quelle est la plus grande tension de sortie d'un CNA si ce dernier fournit 1,0 V quand l'entrée numérique est 00110010?

Solution

$$00110010_2 = 50_{10}$$
$$1{,}0 \text{ V} = K \times 50$$

D'où

$$K = 20 \text{ mV}$$

La plus grande tension que l'on peut obtenir est celle correspondant à l'entrée $1111111_2 = 255_{10}$.

$$V_{SORTIE}(\text{max}) = 20 \text{ mV} \times 255$$
$$= 5{,}10 \text{ V}$$

Sortie analogique D'un point de vue strictement technique, la sortie d'un CNA n'est pas une grandeur analogique, parce qu'elle ne prend que des valeurs spécifiques, comme les 16 niveaux de tension possibles de V_{SORTIE} de la figure 10-2. Dans ce sens, c'est une grandeur numérique. Cependant, comme nous le verrons, le nombre de valeurs de sortie différentes peut être accru et l'écart entre chacune des valeurs peut être rétréci par l'accroissement du nombre de bits d'entrée. Cette possibilité nous permet d'obtenir une sortie qui approche de plus en plus la grandeur analogique qui varie continûment dans une certaine plage de valeurs. En conclusion, la sortie d'un CNA est une grandeur « pseudo-analogique », que nous continuerons tout de même à qualifier d'analogique en gardant à l'esprit, toutefois, que ce n'est qu'une approximation d'une grandeur analogique pure.

Poids de l'entrée Pour le CAN de la figure 10-2, on remarque qu'à chaque chiffre de l'entrée numérique correspond une valeur différente de la sortie analogique. On se convainc

facilement de ceci en examinant les cas où seulement un chiffre de l'entrée est au niveau HAUT:

D	*C*	*B*	*A*		V_{SORTIE}
0	0	0	1	→	1
0	0	1	0	→	2
0	1	0	0	→	4
1	0	0	0	→	8

Les contributions de chaque chiffre numérique sont *pondérées* en fonction de leur rang dans le nombre binaire. Ainsi, A, le bit de poids le plus faible, a un *poids* de 1 V, B a un poids de 2 V, C a un poids de 4 V et D vaut 8 V, c'est le bit de poids le plus fort. Ces poids, à partir du plus faible, doublent à chaque bit. Il est donc possible de considérer que V_{SORTIE} est la somme pondérée des entrées numériques. Par exemple, pour trouver V_{SORTIE} de l'entrée numérique 0111, il suffit d'additionner les poids de C, B et A pour obtenir $4 \text{ V} + 2 \text{ V} + 1 \text{ V} = 7 \text{ V}$.

EXEMPLE 10-2

Soit un convertisseur NA de 5 bits dont $V_{SORTIE} = 0{,}2$ V quand l'entrée numérique est 00001. Trouvez la valeur de V_{SORTIE} si l'entrée est 11111.

Solution À prime abord, on voit que le poids le plus faible est 0,2 V. Donc les poids des autres sont respectivement 0,4 V, 0,8 V, 1,6 V et 3,2 V. Quand l'entrée numérique est 11111, la valeur V_{SORTIE} est $3{,}2 \text{ V} + 1{,}6 \text{ V} + 0{,}8 \text{ V} + 0{,}4 \text{ V} + 0{,}2 \text{ V} = 6{,}2 \text{ V}$.

Résolution (pas de progression) La *résolution* d'un convertisseur NA est définie comme la plus petite variation qui se répercute sur la sortie analogique à la suite d'un changement de l'entrée numérique. Si vous vous reportez au tableau de la figure 10-2, vous pouvez voir que la résolution est 1 V, puisque V_{SORTIE} ne peut changer d'une valeur inférieure à 1 V quand le code d'entrée change. La résolution est toujours égale au poids du bit de poids le plus faible, et on l'appelle **pas de progression**, car c'est l'écart entre les tensions de sortie quand le code d'entrée passe d'un état à son état suivant. C'est ce que montre graphiquement la figure 10-3, dans laquelle les entrées numériques sont fournies par les sorties d'un compteur binaire de 4 bits. Ce compteur est déclenché par un signal d'horloge et parcourt continuellement son cycle de 16 états. La forme d'onde de la sortie NA est un **escalier** de 16 marches, correspondant aux états 0000 à 1111 du compteur, et la différence de tension entre chaque échelon est 1 V. Quand le compteur est à 1111, la sortie du CNA vaut 15 V: la **sortie pleine échelle**. Quand le compteur revient à 0000, la sortie NA retourne à 0 V. La résolution ou le pas de progression est la hauteur des échelons de l'escalier. Dans ce cas-ci, la différence entre chaque échelon est 1 V.

Notez que l'escalier possède 16 marches correspondant aux 16 états d'entrée, alors qu'il n'y a que 15 paliers ou pas entre le niveau 0 V et la pleine échelle. De façon générale, pour un CNA de N bits, le nombre de niveaux différents est de 2^N, alors que le nombre de pas est de $2^N - 1$.

Vous vous êtes sans doute rendu compte que la résolution (hauteur du pas) est identique au facteur de proportionnalité de l'équation entrée/sortie du CNA

$$\text{sortie analogique} = K \times \text{entrée numérique}$$

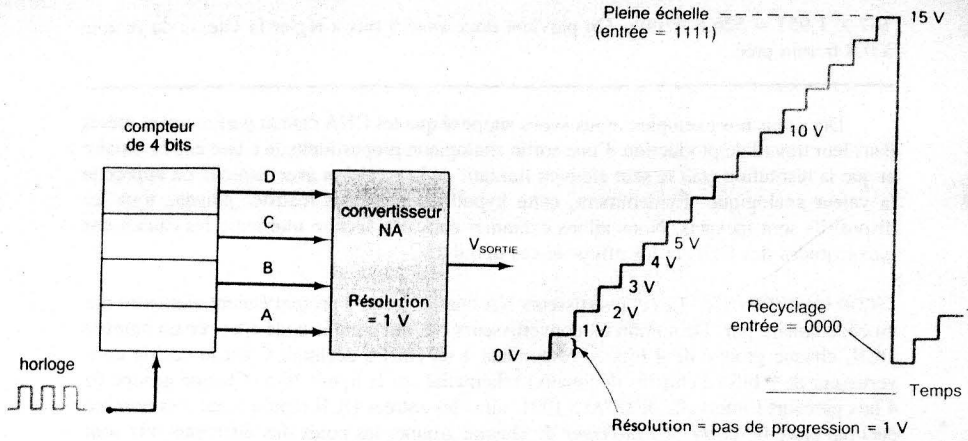


FIGURE 10-3 La forme d'onde de sortie d'un convertisseur NA dont les entrées sont fournies par un compteur binaire.

Une nouvelle interprétation de cette équation est donc de dire que l'entrée numérique est le nombre de pas alors que K est la valeur de la tension (ou du courant) pour un pas. Les exemples suivants illustrent ceci.

EXEMPLE 10-3A

Quelle est la résolution (pas de progression) du CNA de l'exemple 10-2? Décrivez le signal en escalier fourni en sortie par le CNA.

Solution Le poids le plus faible est 0,2 V. Il s'agit de la résolution ou pas de progression. Une forme d'onde en escalier est réalisable si on connecte un compteur de 5 bits aux entrées du CNA. Cet escalier aura 32 marches allant de 0 V à la sortie pleine échelle de 6,2 V, soit 31 paliers de 0,2 V chacun.

EXEMPLE 10-3B

Pour le CNA de l'exemple 10-2, déterminez V_{SORTIE} quand l'entrée numérique est 10001.

Solution Le pas de progression étant 0,2 V, il s'agit du facteur de proportionnalité. L'entrée numérique est 10001, soit l'équivalent de 17_{10}. Donc, la tension de sortie est:

$$V_{SORTIE} = (0{,}2\ \text{V}) \times 17$$
$$= 3{,}4\ \text{V}$$

Résolution (% PE) Certes, il est possible d'exprimer la résolution en indiquant la valeur absolue du changement du courant ou de la tension, mais il est beaucoup plus utile de l'exprimer en pourcentage de la **pleine échelle**. Comme exemple, voyons le convertisseur NA de la figure 10-3 dont la tension pleine échelle est 15 V (quand l'entrée numérique est

1111). La résolution est 1 V, ce qui donne comme résolution exprimée en pourcentage de la pleine échelle:

$$\text{résolution (\% PE)} = \frac{\text{pas de progression}}{\text{pleine échelle}} \times 100\ \% \qquad (10\text{-}2)$$
$$= \frac{1\ \text{V}}{15\ \text{V}} \times 100\ \% = 6{,}67\ \%$$

EXEMPLE 10-4

Un convertisseur NA de 10 bits a un pas de progression de 10 mV. Déterminez la tension pleine échelle et la résolution en pourcentage de la pleine échelle.

Solution Avec 10 bits, on a $2^{10} - 1 = 1023$ pas de 10 mV chacun. La sortie pleine échelle est donc 10 mV × 1023 = 10,23 V et

$$\text{résolution (\% PE)} = \frac{10\ \text{mV}}{10{,}23\ \text{V}} \times 100\% \approx 0{,}1\%$$

L'exemple 10-4 nous aide à comprendre que la résolution exprimée en pourcentage de la pleine échelle devient un nombre plus petit à mesure que le nombre de bits d'entrée augmente. En fait, le pourcentage de la pleine échelle peut aussi être calculé ainsi:

$$\text{résolution (\% PE)} = \frac{1}{\text{nombre total de pas}} \times 100\ \% \qquad (10\text{-}3)$$

Quand le nombre binaire contient N-bits, le nombre total de pas est $2^N - 1$. Donc, pour l'exemple précédent,

$$\text{résolution (\% PE)} = \frac{1}{2^{10} - 1} \times 100\ \%$$
$$= \frac{1}{1023} \times 100\ \%$$
$$\approx 0{,}1\ \%$$

On peut donc déduire de cela que *seul le nombre de bits* détermine le *pourcentage* de la pleine échelle (résolution). En augmentant le nombre de bits, on augmente le nombre d'échelons pour atteindre la pleine échelle, de sorte que chaque échelon devient un élément plus petit de la tension pleine échelle. Dans les spécifications de la plupart des fabricants de CNA, la résolution est exprimée par le nombre de bits.

Définition de la résolution Un CNA ne peut produire une plage continue de valeurs de sortie et c'est la raison pour laquelle, à proprement parler, on ne peut affirmer que sa sortie est une grandeur analogique pure. L'ensemble des valeurs de sortie d'un CNA est donc fini. Dans notre exemple de la section 10-1, l'ordinateur générait une sortie numérique utilisée pour obtenir une tension entre 0 V et 10 V servant à régler l'ouverture d'une soupape électrique. La résolution du CNA (nombre de bits) impose le nombre de valeurs de tension que l'ordinateur pourra envoyer à la soupape. S'il s'agit d'un CNA de 6 bits, le nombre de niveaux sera de 63 et la progression se fera par pas de 0,159 V entre 0 et 10 V. Par ailleurs, dans le cas d'un CNA de 8 bits, le nombre de niveaux passe à 255, et le pas de progression est alors 0,039 V. Plus nous utilisons de bits, plus la résolution est fine (la hauteur du pas est petite).

Le concepteur de système doit décider de la résolution dont il a besoin sur la base de la performance exigée pour le système. La résolution limite la précision avec laquelle la sortie du CNA approche une grandeur analogique donnée. Habituellement, le prix d'un CNA augmente avec son nombre de bits, donc le concepteur devra s'en tenir au plus petit nombre de bits qui donne un résultat acceptable.

EXEMPLE 10-5

La figure 10-4 reproduit le montage d'un ordinateur qui contrôle la vitesse d'un moteur. Le courant entre 0 et 2 mA qui sort du CNA est amplifié afin de permettre au moteur d'atteindre des vitesses entre 0 et 1000 tr/min (tours par minute). Combien faut-il de bits pour que l'ordinateur puisse régler la vitesse du moteur à une vitesse particulière avec une précision de 2 tr/min?

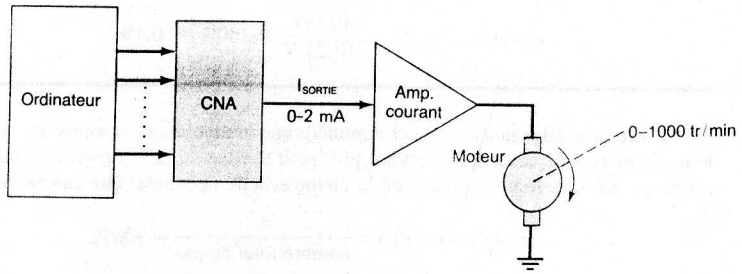


FIGURE 10-4 Exemple 10-5.

Solution La vitesse du moteur parcourt l'intervalle 0 à 1000 tr/min quand la sortie passe de zéro à la pleine échelle. Chaque pas de la sortie du CNA fait progresser la vitesse du moteur. Nous voulons que le pas de progression de la vitesse du moteur ne dépasse pas 2 tr/min. Il nous faut donc 500 paliers (1000/2). Cherchons maintenant le nombre de bits qui nous donne au moins 500 paliers entre 0 et la pleine échelle. Nous savons que le nombre de pas est donné par $2^N - 1$, donc il faut:

$$2^N - 1 \geq 500$$

ou

$$2^N \geq 501$$

Comme $2^8 = 256$ et $2^9 = 512$, le plus petit nombre de bits qui nous procure au moins 500 pas est *neuf*. On pourrait utiliser un plus grand nombre de bits, mais au prix d'une plus grande dépense.

EXEMPLE 10-6

En utilisant 9 bits, dites avec quelle précision on parvient à régler la vitesse de 326 tr/min.

Solution Avec 9 bits, on obtient 511 pas ($2^9 - 1$). Donc, la vitesse du moteur progresse par pas de 1000 tr/min/511 = 1,957 tr/min. Le nombre de pas nécessaire pour atteindre la valeur désirée de 326 tr/min est 326/1,957 = 166,58. Comme ce dernier nombre n'est pas un entier, il faut arrondir à 167. La vitesse réelle du moteur au 167e pas sera

167 × 1,957 = 326,8 tr/min. On parvient donc avec 9 bits à régler la vitesse du moteur à 0,8 tr/min près.

Dans tous nos exemples, nous avons supposé que les CNA étaient parfaitement précis dans leur travail de production d'une sortie analogique proportionnelle à une entrée binaire et que la résolution était le seul élément limitatif de la précision avec laquelle on approche la valeur analogique. Évidemment, cette hypothèse n'est pas réaliste, puisque tous les dispositifs sont inexacts. Nous allons examiner dans une section ultérieure les causes des inexactitudes des CNA et les effets de ces dernières.

Code d'entrée DCB Les convertisseurs NA étudiés jusqu'à présent étaient alimentés par un code binaire pur. De nombreux convertisseurs NA ont comme code d'entrée un nombre DCB, chaque groupe de 4 bits correspondant à un chiffre décimal. C'est le cas du convertisseur de 8 bits (2 chiffres décimaux) schématisé sur la figure 10-5. Chaque groupe de 4 bits parcourt l'intervalle de 0000 à 1001, ainsi les entrées DCB représentent tout nombre décimal entre 00 et 99. *À l'intérieur* de chaque groupe, les poids des différents bits sont identiques à ceux de la représentation binaire pure (1, 2, 4, 8), mais les poids des *groupes* diffèrent par un facteur de 10. Par exemple, A_0, le bit de poids faible du chiffre de poids faible, a un poids de 0,1 V. Voyez à la figure 10-5 les poids relatifs de chaque chiffre. Remarquez que les poids du groupe de poids fort sont dix fois ceux du groupe de poids faible.

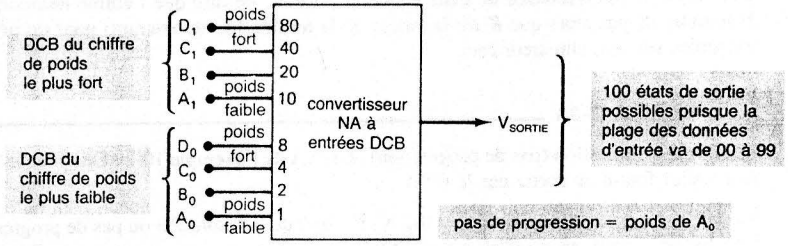


FIGURE 10-5 Convertisseur NA dont l'entrée est un nombre DCB. Ce convertisseur accepte une entrée de 2 chiffres et produit 100 valeurs analogiques différentes.

EXEMPLE 10-7A

Si le poids de A_0 de la figure 10-5 est 0,1 V, trouvez:

a) Le pas de progression.

b) La tension pleine échelle et la résolution exprimée en pourcentage de la pleine échelle.

c) V_{SORTIE} pour $D_1C_1B_1A_1 = 0101$ et $D_0C_0B_0A_0 = 1000$.

Solution

a) Le pas de progression est le poids du bit de poids le plus faible du chiffre de poids le plus faible, soit 0,1 V.

b) Il y a 99 échelons puisqu'il y a deux chiffres DCB. Ainsi, la sortie pleine échelle est 99 × 0,1 = 9,9 V. On trouve donc comme résolution en utilisant l'équation (10-2)

$$\frac{\text{pas de progression}}{\text{PE}} \times 100\% = \frac{0.1}{9{,}9} \times 100\% \approx 1\%$$

On aurait pu utiliser l'équation (10-3) pour calculer la résolution exprimée en pourcentage PE, puisqu'on sait que le nombre total d'échelons est 99.

c) Les poids exacts en volts sont:

CHIFFRE POIDS FORT				CHIFFRE POIDS FAIBLE			
D_1	C_1	B_1	A_1	D_0	C_0	B_0	A_0
8,0	4,0	2,0	1,0	0,8	0,4	0,2	0,1

Pour trouver V_{SORTIE}, additionnez les poids de tous les rangs où la valeur est 1. Cela donne

$$V_{SORTIE} = \overbrace{0.8\text{ V}}^{D_0} + \overbrace{4\text{ V}}^{C_1} + \overbrace{1\text{ V}}^{A_1} = \mathbf{5.8\text{ V}}$$

Voici une autre façon plus facile de faire le calcul: le code d'entrée est 58, il suffit donc de multiplier ce code par 0,1 V ce qui donne de nouveau 5,8 V.

EXEMPLE 10-7B

Un convertisseur numérique-analogique DCB de 12 bits délivre une tension pleine échelle de 9,99 V.

a) Déterminez la résolution en % PE.

b) Déterminez le pas de progression du convertisseur.

Solution

a) Douze bits, en DCB, signifient trois chiffres décimaux ou plus précisément tous les chiffres décimaux entre 000 et 999. Par conséquent, la sortie du convertisseur NA possède 999 niveaux distincts entre 1 et 9,99 V. Il vient:

$$\text{résolution en \%} = \frac{1}{\text{nombre de pas}} \times 100\ \%$$

$$= \frac{1}{999} \times 100\ \%$$

$$\approx \mathbf{0{,}1\ \%}$$

b)
$$\text{pas de progression} = \frac{\text{P.E.}}{\text{nombre de pas}}$$

$$= 9{,}99\text{ V}/999$$

$$= \mathbf{0{,}01\text{ V}}$$

CNA bipolaires Jusqu'à présent, nous avons retenu l'hypothèse que l'entrée binaire d'un CNA était un nombre non signé et que la tension ou le courant de sortie était une valeur positive. Certains CNA sont étudiés pour fournir aussi bien des valeurs positives que négatives, comme de -10 V à $+10$ V. On parvient généralement à obtenir cela en fournissant en entrée un nombre signé dont le bit de rang supérieur est le bit de signe (0 pour + et 1 pour −). Les valeurs d'entrée négatives sont souvent exprimées selon la notation en complément à 2, bien qu'il existe des CNA dont l'entrée accueille un nombre binaire pur. Par exemple, supposons que nous ayons un CNA bipolaire de 6 bits fonctionnant avec la notation en complément à 2 et dont la précision est 0,2 V. La plage des entrées binaires va de 100000 (−32) à 011111 (+31); les tensions de sortie correspondantes vont de −6,4 V à +6,2 V. Il y a 63 pas de progression de 0,2 V entre ces deux bornes extrêmes.

QUESTIONS DE RÉVISION

1. Un CNA de 8 bits a en sortie un courant de 3,92 mA quand son entrée est 01100010. Quelle est la résolution du CNA et sa sortie pleine échelle?
2. Quel est le poids du bit fort du CNA de la question 1?
3. Quelle est la résolution en pourcentage d'un CNA de 8 bits?
4. Combien de tensions de sortie distinctes sont fournies par un CNA de 12 bits?
5. Pour le montage de la figure 10-4, combien de bits faudrait-il pour que l'ordinateur règle la vitesse avec une précision de 0,4 tr/min?
6. Soit un CNA de 12 bits dont les entrées sont des chiffres DCB et la résolution est 10 mV. Quelle sortie correspondra à une entrée de 100001110011?
7. *Vrai ou faux:* La résolution en % PE d'un CNA dépend *essentiellement* du nombre de bits.
8. Quel est l'avantage d'avoir une résolution plus fine?

10-3 CIRCUITS DES CONVERTISSEURS NA

Il existe plusieurs méthodes et circuits pour matérialiser la conversion NA que l'on vient de décrire. Nous nous bornons ici à étudier quelques méthodes élémentaires, afin de vous donner une idée des principes utilisés. Il n'est pas important que vous connaissiez en détails les divers types de circuits, car les convertisseurs NA sont vendus en version intégrée, ou encapsulés dans des boîtiers dont l'utilisation n'exige pas que vous connaissiez les circuits. Au contraire, il est préférable que vous connaissiez seulement de façon générale les caractéristiques de rendement importantes des convertisseurs NA afin de savoir les utiliser intelligemment. L'étude de ces caractéristiques fait l'objet de la section 10-4.

La figure 10-6 a) nous montre le circuit de base d'un type de convertisseur NA de 4 bits. Les entrées A, B, C et D sont des entrées binaires qui, par hypothèse, prennent la valeur 0 V ou 5 V. L'*amplificateur opérationnel* est employé comme amplificateur de sommation dont la sortie est égale à la somme pondérée des tensions d'entrée. Rappelons qu'un amplificateur de sommation multiplie chaque tension de sortie par le rapport de la résistance de contre-réaction R_F à la résistance d'entrée correspondante $R_{ENTRÉE}$. Dans ce circuit, $R_F = 1$ kΩ et les résistances d'entrée vont de 1 à 8 kΩ. L'entrée D a $R_{ENTRÉE} = 1$ kΩ, de sorte que l'amplificateur de sommation communique la tension appliquée à D sans affaiblissement. L'entrée C a $R_{ENTRÉE} = 2$ kΩ, de sorte que le facteur d'affaiblissement est 0,5. De même, l'entrée B est affaiblie d'un facteur de 0,25 et l'entrée A d'un facteur 0,125. L'expression de sortie de l'amplificateur est donc

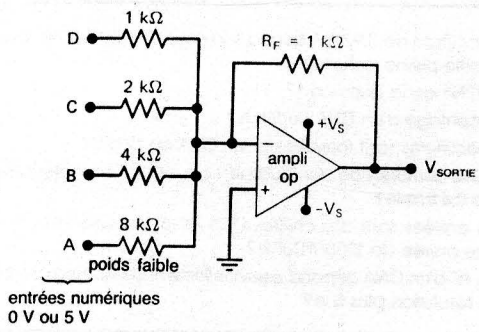
$$V_{SORTIE} = -(V_D + 0{,}5V_C + 0{,}25V_B + 0{,}125V_A) \qquad (10\text{-}4)$$

Le signe moins à l'avant des parenthèses est ajouté parce que l'amplificateur de sommation est un amplificateur inverseur; évidemment ceci n'a aucun effet ici.

De cette façon, il est facile de voir que la sortie de l'amplificateur de sommation est une tension analogique qui représente la somme pondérée des entrées numériques, comme l'indique le tableau de la figure 10-6 b). Ce tableau énumère toutes les conditions d'entrée possibles et la tension de sortie résultante de l'amplificateur. La sortie est calculée, pour une condition d'entrée quelconque, en mettant les entrées pertinentes soit à 0 V, soit à 5 V. Par exemple, si l'entrée numérique est 1010, alors $V_D = V_B = 5$ V et $V_C = V_A = 0$ V. Il vient donc d'après l'équation (10-4)

$$V_{SORTIE} = -(5\text{ V} + 0\text{ V} + 0{,}25 \times 5\text{ V} + 0\text{ V})$$

$$= -6{,}25\text{ V}$$



a)

code d'entrée

D	C	B	A	V_{SORTIE} (volts)	
0	0	0	0	0	
0	0	0	1	−0,625	← poids faible
0	0	1	0	−1,250	
0	0	1	1	−1,875	
0	1	0	0	−2,500	
0	1	0	1	−3,125	
0	1	1	0	−3,750	
0	1	1	1	−4,375	
1	0	0	0	−5,000	
1	0	0	1	−5,625	
1	0	1	0	−6,250	
1	0	1	1	−6,875	
1	1	0	0	−7,500	
1	1	0	1	−8,125	
1	1	1	0	−8,750	
1	1	1	1	−9,375 ←	pleine échelle

b)

FIGURE 10-6 Convertisseur NA élémentaire utilisant un amplificateur opérationnel de sommation.

La résolution de ce convertisseur NA est le poids le plus faible, soit 0,125 × 5 V = 0,625 V. Comme le montre le tableau, les sorties analogiques croissent par pas de 0,625 V quand le nombre binaire d'entrée avance d'un bit.

EXEMPLE 10-8

a) Déterminez le poids de chacun des bits d'entrée de la figure 10-6 a).
b) Substituez à R_F la valeur 250 Ω et déterminez la tension pleine échelle.

Solution

a) Le bit de poids le plus fort traverse intégralement l'amplificateur et son gain = 1, de sorte que son poids à la sortie est 5 V. Donc

poids le plus fort → 5 V

2e poids fort → 2,5 V

3e poids fort → 1,25 V

4e poids fort = poids faible → 0,625 V

b) Si R_F est réduit par un facteur de 4 jusqu'à 250 Ω, chaque poids d'entrée est 4 fois *plus petit* que la valeur précédente. Donc la tension pleine échelle se trouve réduite par le même facteur et devient −9,375/4 = −2,344 V.

Une observation attentive des résistances de la figure 10-6 nous apprend que leurs valeurs progressent comme l'inverse **des poids des rangs binaires**. Autrement dit, en commençant par le bit de poids fort, on constate que la résistance double d'une branche à l'autre. Ceci évidemment produit la pondération recherchée pour la tension de sortie.

Précision de la conversion Le tableau de la figure 10-6 b) nous fait connaître les valeurs *idéales* de V_{SORTIE} pour les différents cas d'entrée. L'exactitude avec laquelle ce circuit reproduit ces valeurs dépend de deux facteurs: 1) de la précision des résistances d'entrée et de contre-réaction, et 2) de la précision des niveaux de tension d'entrée. On peut obtenir des résistances très précises (à 0,01 pourcent des valeurs recherchées) par ajustement; pour ce qui est des niveaux de tension d'entrée, ce n'est pas aussi facile. D'abord, il est bien évident que les entrées numériques ne peuvent être prélevées directement de sorties de bascules ou de portes logiques, parce que les niveaux logiques de sortie de ces dispositifs ne sont pas exactement 0 V et 5 V, puisqu'ils varient dans une plage donnée. Pour cette raison, il est nécessaire d'intercaler des éléments de circuit entre chaque borne numérique et sa résistance d'entrée connectée à l'amplificateur de sommation, comme on peut le constater à la figure 10-7.

Chaque entrée numérique agit sur un interrupteur à semiconducteur comme l'interrupteur bidirectionnel CMOS étudié au chapitre 8. Au moment où l'entrée est au niveau HAUT, l'interrupteur se ferme et raccorde une *alimentation de référence précise* à la résistance d'entrée; quand l'entrée est BASSE, l'interrupteur s'ouvre. L'alimentation de référence délivre une tension précise et très stable, ce qui est indispensable si on veut obtenir une sortie analogique exacte.

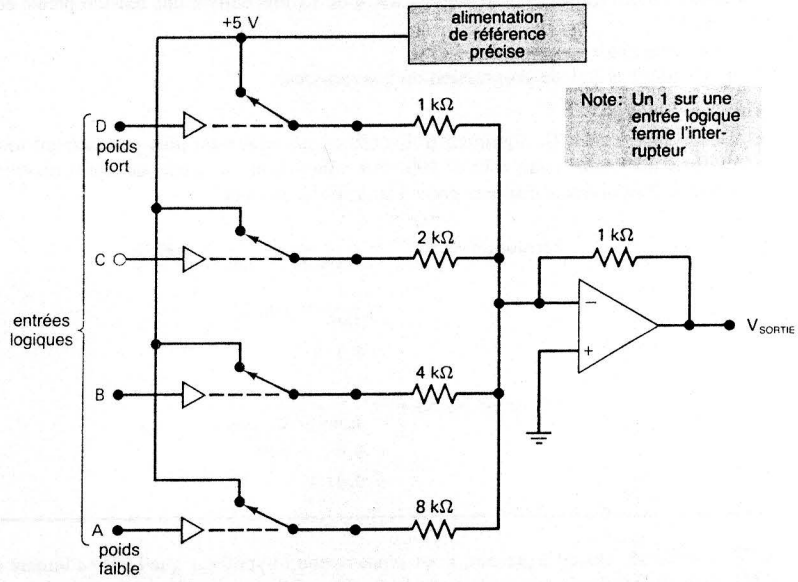


FIGURE 10-7 Convertisseur NA complet de 4 bits avec amplificateurs de niveau de précision.

CNA dont la sortie est un courant Le schéma élémentaire d'une méthode de production d'un courant de sortie analogique proportionnel à l'entrée binaire est illustré à la figure 10-8 a). Le circuit montré est un CNA 4 bits possédant des résistances dont les valeurs correspondent aux poids d'un nombre binaire. Il y a dans ce circuit quatre trajets de courant parallèles, chacun étant contrôlé par un interrupteur à semiconducteur comme l'interrupteur bilatéral CMOS. L'état de chaque interrupteur est commandé par les niveaux de tension appliqués aux entrées binaires. Le courant qui traverse chaque trajet parallèle est déterminé par V_{REF} et la résistance du trajet. La pondération de ces résistances est

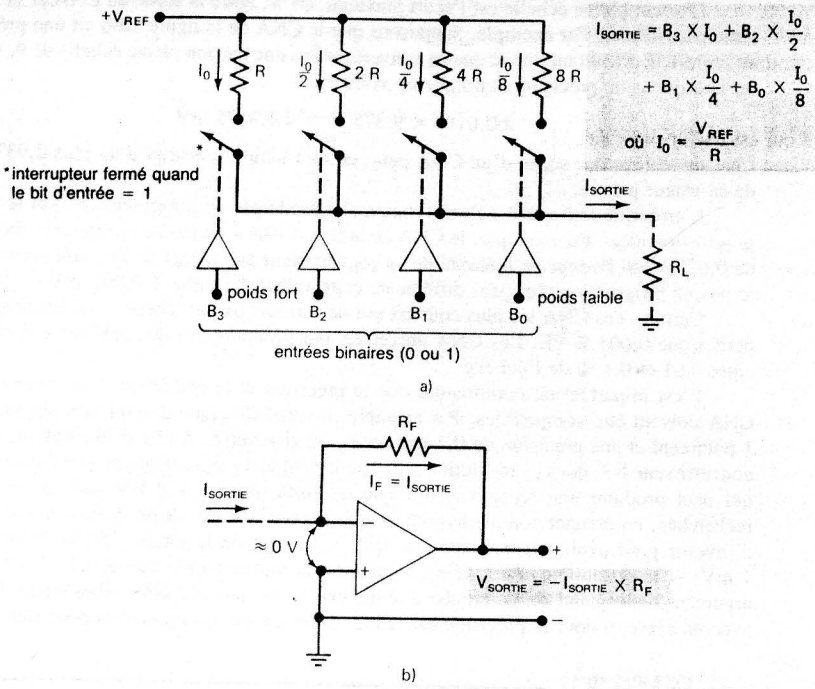


FIGURE 10-8 a) CNA élémentaire dont la sortie est un courant; b) raccordé à un ampli op agissant comme convertisseur courant-tension.

identique à celle de la suite binaire, et le courant total, I_{SORTIE}, est fonction des interrupteurs ouverts ou fermés (des 0 et des 1 des entrées binaires). La résistance du trajet poids fort est la plus petite, R; la résistance suivante est le double; et ainsi de suite. On peut dériver le courant de sortie à travers une charge, R_L, mais celle-ci doit être beaucoup plus petite que R afin de laisser presque inchangée l'intensité du courant. Idéalement, R_L doit être un court-circuit avec la masse.

EXEMPLE 10-9

Supposez $V_{REF} = 10$ V et $R = 10$ kΩ. Déterminez la résolution et le courant pleine échelle de ce CNA.

Solution $I_0 = V_{REF}/R = 1$ mA. Il s'agit là du poids le plus fort. Les trois autres courants sont 0,5, 0,25 et 0,125 mA. Le poids le plus faible est 0,125 mA, ce poids correspond également à la résolution.

L'intensité pleine échelle est obtenue quand les entrées binaires sont toutes au niveau HAUT et que tous les interrupteurs de courant sont fermés, ceci donne

$$I_{SORTIE} = 1 + .5 + .25 + .125 = \mathbf{1.875\ mA}$$

Notez que le courant de sortie est proportionnel à V_{REF}. Si V_{REF} augmente ou diminue, I_{SORTIE} change proportionnellement.

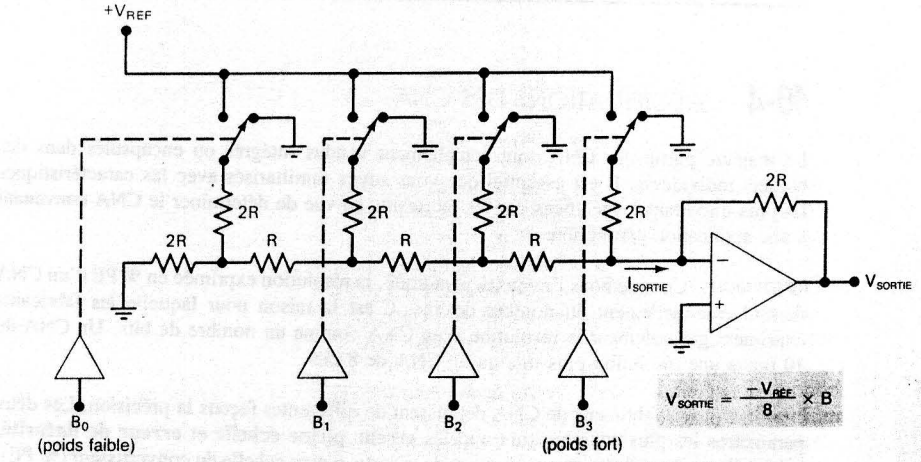
Pour que I_{SORTIE} soit précis, il faudrait que R_L soit un court-circuit avec la masse. Une façon de réaliser quand même ceci est de recourir à un amplificateur opérationnel agissant comme convertisseur courant-tension, comme on le montre à la figure 10-8 b). Dans un tel cas, I_{SORTIE} du CNA est connecté à l'entrée « − » de l'amplificateur opérationnel, entrée qui est une masse virtuelle. La réaction négative de l'amplificateur opérationnel produit un courant égal à I_{SORTIE} à travers R_F pour donner $V_{SORTIE} = -I_{SORTIE} \times R_F$. Ainsi, V_{SORTIE} est une tension analogique proportionnelle à l'entrée binaire du CNA. Cette sortie analogique peut attaquer une grande variété de charges sans subir de diminution.

Réseau R/2R Les circuits des convertisseurs numériques-analogiques que nous venons d'étudier exploitent une suite de résistances pondérées comme la suite binaire afin d'obtenir pour chaque bit le poids approprié. Bien que cette méthode soit excellente en théorie, en pratique, elle impose certaines contraintes. La plus importante étant la grande différence entre les deux résistances extrêmes (celle du bit de poids fort et celle du bit de poids faible), particulièrement dans le cas des CNA de grande résolution (à plusieurs bits). Par exemple, la résistance du poids fort pour un CNA de 12 bits est 1 kΩ, tandis que celle du bit de poids faible pourra dépasser les 2 MΩ. Dans l'état actuel de la technologie de fabrication des CI, il est très difficile de réaliser des résistances aussi éloignées qui conservent un rapport précis, particulièrement en présence de variations de température.

Pour cela, il est préférable d'utiliser un circuit dans lequel les résistances ont des valeurs voisines. L'une des conceptions les plus courantes réalisant ce souhait est le *réseau R/2R*, où les résistances ne sont que dans un rapport de 2 à 1. On peut voir l'un de ces CNA à la figure 10-9.

Remarquez le montage des résistances, particulièrement les deux valeurs différentes employées: R et $2R$. Le courant I_{SORTIE} dépend des positions des quatre interrupteurs et ce sont les entrées binaires $B_3B_2B_1B_0$ qui contrôlent les états des interrupteurs. Ce courant est transmis à un convertisseur courant-tension (en fait un amplificateur opérationnel) qui

FIGURE 10-9 CNA réseau *R/2R* de base.



délivre la tension V_{SORTIE}. Nous n'analyserons pas en détail ce circuit; il suffit simplement de dire qu'on peut démontrer que V_{SORTIE} est donnée par l'expression:

$$V_{SORTIE} = \frac{-V_{RÉF}}{8} \times B \qquad (10\text{-}5)$$

où B est la valeur de l'entrée binaire qui s'échelonne entre 0000 (0) et 1111 (15).

EXEMPLE 10-10

Présumons que $V_{RÉF} = 5$ V pour le CNA de la figure 10-9. Quelle est la résolution et quelle est la tension pleine échelle de ce convertisseur?

Solution La résolution est égale au poids le plus faible, c'est-à-dire à la tension quand $B = 0001 = 1$ dans l'équation (10-5):

$$\text{résolution} = \frac{-5 \text{ V} \times 1}{8}$$
$$= -0{,}625 \text{ V}$$

La sortie pleine échelle est la tension correspondant à l'entrée $B = 1111 = 15_{10}$. De nouveau, en recourant à l'équation (10-5), il vient:

$$\text{pleine échelle} = \frac{-5 \text{ V} \times 15}{8}$$
$$= -9{,}375 \text{ V}$$

QUESTIONS DE RÉVISION

1. Quel est l'avantage d'un CNA réseau *R/2R* par rapport à un convertisseur à résistances pondérées binairement?
2. Un certain CNA de 6 bits fonctionne au moyen de résistances pondérées. Si la résistance du bit de poids fort est 20 kΩ, quelle est la résistance du bit de poids faible.
3. Quelle est la résolution si la valeur de R_f dans la figure 10-6 est changée pour 800 Ω?

10-4 SPÉCIFICATIONS DES CNA

La majeure partie des CNA sont actuellement vendus intégrés ou encapsulés dans des boîtiers individuels. Il est essentiel que vous soyez familiarisés avec les caractéristiques les plus importantes spécifiées par les fabricants en vue de déterminer le CNA convenant à une application particulière.

Résolution Comme nous l'avons déjà indiqué, la résolution exprimée en % PE d'un CNA dépend essentiellement du nombre de bits. C'est la raison pour laquelle les fabricants expriment généralement la résolution d'un CNA comme un nombre de bits. Un CNA de 10 bits a une résolution plus fine qu'un CNA de 8 bits.

Précision Les fabricants de CNA définissent de différentes façons la précision. Les deux paramètres les plus courants sont appelés **erreur pleine échelle** et **erreur de linéarité**, habituellement exprimés en pourcentage de la sortie pleine échelle du convertisseur (% PE).

L'erreur pleine échelle est l'écart maximal, en %, entre la sortie du CNA et sa valeur anticipée (idéale). Par exemple, supposons que le CNA de la figure 10-6 ait une précision de ±0,01 % PE. Étant donné que ce convertisseur a une tension pleine échelle de 9,375 V, on voit que cette précision exprimée en volts est:

$$\pm 0{,}01\% \times 9{,}375 \text{ V} = \pm 0{,}9375 \text{ mV}$$

Ceci signifie qu'une sortie d'un CNA peut, en tout temps, s'écarter d'au plus 0,9375 mV de sa valeur prévue.

L'erreur de linéarité est l'écart maximal entre le pas de progression réel et le pas de progression idéal. Par exemple, le CNA de la figure 10-6 a un pas de progression théorique de 0,625 V. Si l'erreur de linéarité de ce convertisseur est ±0,01 % PE, cela signifie que le *pas de progression* réel peut différer de cette valeur d'au plus 0,00625 mV.

Certains des CNA les plus coûteux ont des erreurs pleine échelle et de linéarité aussi petites que 0,001 % PE. Les CNA universels ont généralement des précisions comprises entre 0,01 et 0,1 % de l'échelle.

Il est important de comprendre que la précision et la résolution d'un convertisseur CNA doivent être compatibles. Par exemple, il serait illogique d'avoir une résolution de 1 pour cent et une précision de 0,1 pour cent, ou vice versa. À titre d'illustration, soit un convertisseur NA dont la résolution est 1 pour cent et la tension pleine échelle de 10 V, qui peut produire une tension analogique de sortie précise à 0,1 V près de la valeur recherchée, en prenant comme hypothèse que la précision est idéale. Cela n'a aucun sens d'investir pour avoir une précision de 0,01 pour cent de la pleine échelle (c'est-à-dire 1 mV) si la résolution limite déjà l'écart avec la valeur recherchée à 0,1 V. Le même argument tient quand on recherche à avoir une résolution très petite (beaucoup de bits) avec un appareil dont la précision est faible; on utilise beaucoup de bits pour rien.

EXEMPLE 10-11

Un CNA de 8 bits possède une sortie pleine échelle de 2 mA et une erreur pleine échelle de ±0,5 %. Quelle est la plage des sorties possibles correspondant à l'entrée 10000000?

Solution Le pas de progression est 2 mA/255 = 7,84 μA. Comme $10000000 = 128_{10}$, la sortie idéale devrait être 128 × 7,84 μA = 1004 μA. Comme l'erreur peut aller jusqu'à:

$$\pm 0{,}5\ \% \times 2 \text{ mA} = \pm 10\ \mu\text{A}$$

Il s'ensuit que la sortie réelle peut s'écarter de la valeur idéale de 1004 μA de cette quantité; la plage des sorties possibles est dès lors 994 à 1114 μA.

Temps d'établissement La vitesse de fonctionnement d'un CNA est définie par son *temps d'établissement*, qui est le temps nécessaire à la sortie pour passer de 0 à la tension pleine échelle quand l'entrée binaire change tous ses 0 pour des 1. En réalité, le temps d'établissement est défini comme le temps que prend le CNA pour se stabiliser à ±1/2 pas de progression (résolution) de sa valeur finale. Par exemple, si un CNA a une résolution de 10 mV, le temps d'établissement est la mesure du temps qui s'écoule avant que la sortie se stabilise à 5 mV près de sa valeur pleine échelle. Généralement, les temps d'établissement s'échelonnent entre 50 ns et 10 μs. Le plus souvent, les CNA dont la sortie est un courant ont des temps d'établissement plus courts que ceux dont la sortie est une tension. Par exemple le DAC 1280 peut avoir une sortie qui est soit un courant, soit une tension. Son temps d'établissement est de 300 ns quand sa sortie est un courant, et de 2,5 μs quand sa sortie est une tension. La principale raison expliquant cette différence est le temps de réponse de l'amplificateur opérationnel qui convertit le courant en tension.

Tension de décalage Idéalement, la sortie d'un CNA affiche 0 volt quand l'entrée binaire est constituée essentiellement de 0. En pratique, on mesure une petite tension de sortie due à l'**erreur de décalage** de l'amplificateur opérationnel de sortie. Cette erreur de décalage, si elle n'est pas compensée, s'additionne à la valeur prévue de la sortie du CNA, et cela pour *toutes* les entrées possibles. Ainsi, supposons un CNA de 4 bits ayant une erreur de décalage de ± 2 mV et un pas de progression *exact* de 100 mV. Le tableau ci-dessous compare la sortie idéale et la sortie réelle du CNA pour quelques entrées. Notez que la sortie réelle est toujours supérieure de 2 mV à la valeur prévue; cela s'explique par l'erreur de décalage. Cette erreur peut être aussi bien négative que positive.

Code d'entrée	Sortie idéale (mV)	Sortie réelle (mV)
0000	0	2
0001	100	102
1000	800	802
1111	1500	1502

Nombre de CNA possèdent un bouton de réglage du zéro qui permet de ramener à zéro le décalage. On procède généralement à cette compensation en appliquant au CNA une entrée nulle (que des 0) en examinant la sortie et en réglant le *potentiomètre de compensation du décalage* jusqu'à ce que la sortie soit aussi zéro.

Monotonicité Un CNA est dit *monotonique* si sa sortie augmente quand la donnée d'entrée binaire passe d'un palier à l'autre. Une autre façon de décrire cette caractéristique consiste à dire que la sortie en escalier ne comporte aucune marche de descente pendant que l'entrée parcourt les valeurs de zéro à la pleine échelle.

QUESTIONS DE RÉVISION

1. Définissez erreur pleine échelle.
2. Que signifie temps d'établissement?
3. Décrivez l'erreur de décalage et son effet sur la sortie d'un CNA.
4. Pourquoi les CNA dont la sortie est une tension sont-ils généralement plus lents que les CNA dont la sortie est un courant?

10-5 APPLICATIONS DES CNA

Des CNA sont utilisés chaque fois qu'il faut transformer la sortie numérique d'un circuit en une tension ou un courant analogique pour pouvoir entraîner un dispositif analogique. Voici certaines des applications les plus courantes de ces convertisseurs.

Régulation La sortie numérique d'un ordinateur est convertie en un signal de régulation analogique afin de régler la vitesse d'un moteur, la température d'un four ou de commander une variable physique quelconque.

Essai automatique Il est possible de programmer les ordinateurs pour qu'ils produisent les signaux analogiques (par l'entremise d'un CNA) nécessaires à l'essai des circuits analogiques. La réponse analogique du circuit testé est normalement convertie en une valeur numérique par un CAN et réintroduite dans l'ordinateur pour être mémorisée, affichée et parfois analysée.

Reconstitution du signal Dans plusieurs applications, la **numérisation** d'un signal analogique se traduit par la conversion de points successifs en leurs équivalents numériques et la mémorisation des résultats. Cette conversion est réalisée par un convertisseur analogique-numérique (CAN). Un CNA intervient ensuite pour convertir les données numériques en mémoire en un signal analogique — un point à la fois; cet appareil reconstitue pour ainsi dire le signal original. Cet enchaînement de numérisations et de reconstitutions est courant dans les oscilloscopes numériques à mémoire, les lecteurs de disques compacts et les appareils d'enregistrement audio sur bande. Nous en apprendrons plus sur ceci au moment de notre étude des CAN.

Conversion AN Plusieurs types de convertisseurs AN contiennent dans leurs circuits des CNA. C'est ce que nous allons étudier à la section 10-7.

10-6 DÉPANNAGE DES CNA

Les CNA sont de nature hybrique: il sont à la fois analogique et numérique. Il nous faut donc des sondes logiques et des pulseurs pour surveiller les entrées numériques et un multimètre ou un oscilloscope pour évaluer la sortie analogique. Il y a fondamentalement deux façons de tester le fonctionnement d'un CNA: un *test statique de l'exactitude* et un *test de l'escalier*.

Le test statique implique le réglage de l'entrée binaire sur une valeur fixe et la mesure de la sortie analogique avec un appareil de mesure de précision. Ce test permet de s'assurer que la valeur de sortie est bien comprise dans la plage prévue indiquée dans la fiche technique de l'appareil. Si ce n'est pas le cas, il faut en rechercher la raison parmi plusieurs causes; en voici quelques-unes:

- Dérive de la valeur des composants du CNA (résistances) imputable à la température, au vieillissement, ou à d'autres facteurs. Cette cause peut facilement donner lieu à des sorties qui débordent la plage de précision acceptable.
- Connexions ouvertes ou en court-circuit dans l'une des bornes d'entrée. Une telle cause peut empêcher une entrée de contribuer à la sortie analogique ou bien peut toujours additionner une entrée à la sortie analogique. Une défectuosité de cette nature est particulièrement difficile à dépister quand elle survient dans les entrées ayant les poids les plus faibles.
- Une référence de tension erronée. Étant donné que la sortie analogique est rapportée à une tension de référence, on peut obtenir des sorties totalement aberrantes. Pour les CNA dotés de sources de référence extérieures, la vérification de la tension de référence se fait sans difficulté avec un voltmètre numérique. Beaucoup de CNA ont toutefois une source interne qu'il est impossible de contrôler, sauf pour certains dont la tension de référence est sortie sur une broche du boîtier.
- Une erreur de décalage excessive due au vieillissement des composants où à une température inappropriée. Cela occasionne des sorties qui s'écartent de la valeur réelle d'une quantité fixe. Si le dispositif possède un bouton de compensation externe, il est aisé de ramener à zéro une erreur de ce type; cependant, un changement de température pourra faire réapparaître une erreur de décalage.

Le test en escalier a pour rôle de constater la monotonicité du CNA, c'est-à-dire de vérifier que la sortie croît pas à pas à mesure que la donnée d'entrée passe d'une valeur à

l'autre (voir figure 10-3). La hauteur du pas doit toujours être la même et il ne doit pas y avoir de recul ou de marches manquantes avant d'arriver à la valeur pleine échelle. Ce test peut aider à déceler des défaillances internes ou externes qui empêchent une borne d'entrée d'apporter sa contribution au résultat, ou, au contraire, qui imposent en permanence le poids d'une borne d'entrée. Voici un exemple illustrant ce test.

EXEMPLE 10-12

Quel sera l'aspect de la forme d'onde dans un test en escalier si le CNA de la figure 10-3 a son entrée *C* ouverte? Supposez que les entrées CNA sont compatibles TTL.

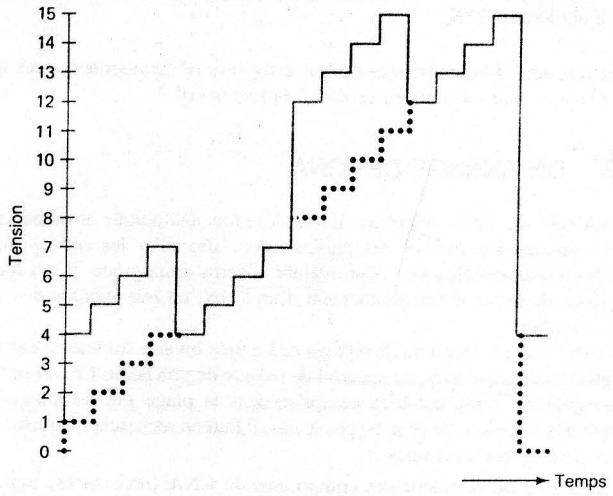


FIGURE 10-10 Exemple 10-12.

Solution Une connexion *C* ouverte est interprétée comme un 1 logique constant par le CNA. Donc, cette entrée ajoute toujours 4 V à la sortie du CNA, ce qui donne une forme d'onde ayant l'aspect montré à la figure 10-10. Les lignes en pointillées dessinent l'escalier tel qu'il devrait être pour le CNA en bon état de marche. Notez que la forme d'onde de sortie coïncide avec la forme d'onde correcte quand les valeurs des entrées ont le bit *C* au niveau HAUT.

10-7 CONVERSION ANALOGIQUE-NUMÉRIQUE

Un *convertisseur AN* reçoit une tension d'entrée analogique qu'il convertit, après un certain temps, en un code de sortie numérique correspondant à l'entrée. La conversion AN est généralement plus complexe et plus longue que la conversion NA. Différentes méthodes de conversion AN ont été mises au point et sont utilisées. Nous allons consacrer le reste de cette section à l'étude détaillée de quelques méthodes, bien qu'il y ait fort à parier que vous n'aurez jamais à concevoir ou à construire de convertisseurs AN (qui sont vendus

complètement montés dans des boîtiers). Toutefois, la connaissance des techniques utilisées vous fera prendre conscience des facteurs qui déterminent le rendement d'un convertisseur AN.

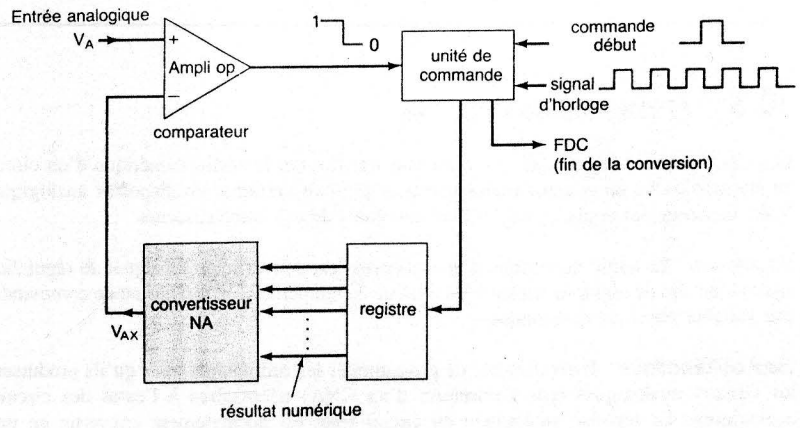
On retrouve dans les circuits de plusieurs types de CAN un convertisseur NA. La figure 10-11 nous montre le schéma fonctionnel général de cette catégorie de CAN. La synchronisation de la conversion est assurée par un signal d'horloge d'entrée. L'unité de commande renferme les circuits logiques qui génèrent la séquence appropriée d'opérations après l'arrivée de la COMMANDE DÉBUT qui initie le processus de conversion. Le comparateur (un ampli op) a deux entrées *analogiques* et une sortie *numérique* qui prend un état ou l'autre, selon l'entrée analogique qui est la plus grande.

Voici le fonctionnement de base des convertisseurs AN de cette catégorie:

1. La COMMANDE DÉBUT lance la conversion.
2. À une cadence déterminée par le signal d'horloge, l'unité de commande modifie sans arrêt le nombre binaire mémorisé dans le registre.
3. Le nombre binaire du registre est converti en une tension analogique V_{AX} par le convertisseur NA.
4. Le comparateur compare V_{AX} avec l'entrée analogique V_A. Tant que $V_{AX} < V_A$, la sortie du comparateur demeure au niveau HAUT. Quand V_{AX} dépasse V_A d'une valeur au moins égale à V_T (tension seuil), la sortie du comparateur passe au niveau BAS et met fin au processus de progression du nombre dans le registre. À ce point, V_{AX} est une très bonne approximation de V_A et le nombre numérique qui se trouve dans le registre, qui est l'équivalent numérique de V_{AX}, est également l'équivalent numérique de V_A, à l'intérieur des limites de résolution et de précision propres à ce système.
5. La logique de commande valide le signal FDC, fin de conversion, quand la conversion est achevée.

Les diverses variantes de cette solution de conversion NA se distinguent surtout par la façon dont la section de commande modifie sans cesse les nombres dans le registre. À part cela, le principe de base est le même, et la sortie numérique recherchée est mémorisée dans le registre quand la conversion est achevée.

FIGURE 10-11 Schéma général d'une catégorie de convertisseurs AN.



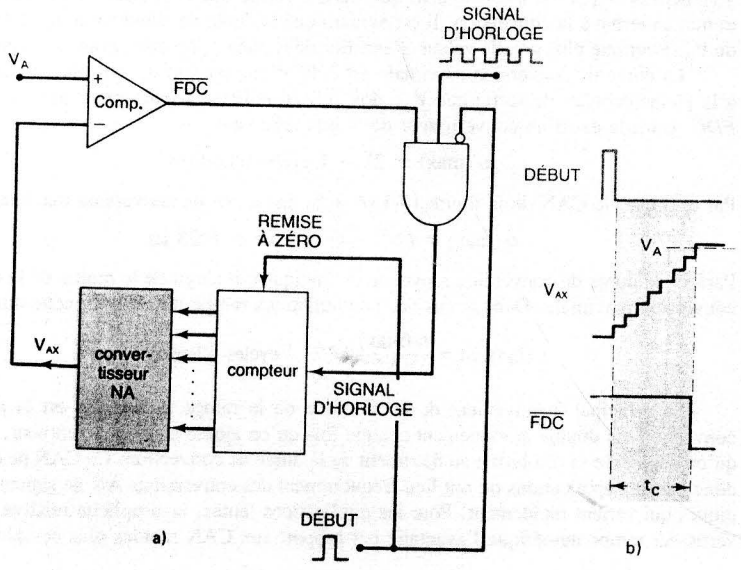
10-8 CAN RAMPE NUMÉRIQUE

L'une des versions les plus simples du convertisseur AN général de la figure 10-11 utilise comme registre un compteur binaire qui est incrémenté par le signal d'horloge (un pas de progression à la fois) jusqu'à ce que $V_{AX} \geq V_A$. On l'appelle *CAN rampe numérique* à cause de la forme d'onde de V_{AX} qui est celle d'une rampe pas à pas (en réalité un escalier), comme celle illustrée à la figure 10-3.

La figure 10-12 reproduit le schéma complet d'un convertisseur en rampe, composé d'un compteur, d'un CNA, d'un comparateur analogique et d'une porte ET. Le comparateur fournit en sortie un signal FDC, vrai au niveau BAS, qui indique la fin de la conversion. Si on suppose que V_A, la tension analogique à convertir est positive, le tout fonctionne comme ceci:

1. On applique une impulsion DÉBUT qui a pour effet de mettre à zéro le compteur et de bloquer la porte ET qui empêche le passage des impulsions d'horloge jusqu'au compteur.
2. Avec que des 0 en entrée, la sortie du CNA sera $V_{AX} = 0$ V.
3. Comme $V_A > V_{AX}$, la sortie du comparateur, $\overline{FDC}$, sera au niveau HAUT.
4. Au moment où DÉBUT revient à 0, la porte ET est validée et les impulsions d'horloge arrivent jusqu'au compteur.
5. À mesure que le compteur progresse, la sortie du CNA, V_{AX}, augmente d'un palier à la fois, comme nous le montre la figure 10-12 b).
6. Ce travail se poursuit tant que V_{AX} n'a pas dépassé V_A d'une quantité au moins égale à V_T (de façon caractéristique 10 à 100 μV). À ce moment-là, $\overline{FDC}$ passe au niveau BAS et bloque le passage du signal d'horloge vers le compteur; ce dernier cesse de compter.

FIGURE 10-12 Convertisseur AN rampe numérique.



7. Le processus de conversion est désormais achevé comme le signale la transition HAUT-BAS de $\overline{FDC}$; le contenu de compteur est la représentation décimale de V_A.
8. Le compteur conserve cette valeur numérique jusqu'à la prochaine impulsion DÉBUT qui déclenche une nouvelle conversion.

EXEMPLE 10-13A

Soit les valeurs suivantes pour le convertisseur AN de la figure 10-12: fréquence de l'horloge = 1 MHz; V_T = 0,1 mV; tension PE du convertisseur NA = 10,23 V et entrée 10 bits. Déterminez:

a) l'équivalent décimal de V_A = 3,728 V;
b) la durée de la conversion; et
c) la résolution de ce convertisseur.

Solution

a) Le CNA a une entrée de 10 bits et une tension PE de 10,23 V. Donc le nombre total de pas possibles est $2^{10} - 1 = 1023$, de sorte que le pas de progression est

$$\frac{10{,}23 \text{ V}}{1023} = 10 \text{ mV}$$

Cela veut donc dire que V_{AX} augmente par pas de 10 mV à mesure que le compteur progresse à partir de 0. Comme V_A = 3,728 V et que V_T = 0,1 mV, V_{AX} doit atteindre 3,7281 V ou une valeur supérieure pour que le comparateur passe à 0. Le nombre de pas nécessaire est:

$$\frac{3{,}7281 \text{ V}}{10 \text{ mV}} = 372{,}81 = 373 \text{ pas}$$

À la fin de la conversion, le nombre mémorisé dans le compteur sera l'équivalent binaire de 373, c'est-à-dire 0101110101. Il s'agit de l'équivalent numérique recherché pour V_A = 3,728 V que produit le convertisseur AN.

b) Il faut 373 pas pour mener à bien cette conversion. Donc, 373 impulsions d'horloge dont la période est une microseconde donne une durée de conversion totale de 373 μs.

c) La résolution de ce convertisseur est égale au pas de progression du convertisseur NA, c'est-à-dire 10 mV. Exprimé en pourcentage de la pleine échelle, cela donne 10 mV/10,23 V × 100 pour cent 0,1 pour cent.

EXEMPLE 10-13B

Pour le même CAN de l'exemple 10-13A, déterminez la plage approximative des tensions analogiques qui donne toujours le même résultat numérique, soit $0101110101_2 = 373_{10}$.

Solution Le tableau suivant illustre la tension de sortie idéale V_{AX} du CNA pour plusieurs pas de progression de part et d'autre du 373e pas.

Étape	V_{AX}
371	3,71 V
372	3,72 V
373	3,73 V
374	3,74 V
375	3,75 V

Si V_A est légèrement inférieur à 3,72 V (par une quantité V_T), alors $\overline{FDC}$ ne passe pas au niveau BAS quand V_{AX} atteint le palier 3,72 V, mais plutôt quand il atteint le palier 3,73 V. Si V_A est légèrement inférieur à 3,73 V (par une quantité V_T), alors $\overline{FDC}$ ne passe pas au niveau BAS tant que V_{AX} n'atteint pas le palier 3,74 V. Ainsi, tant que V_A est approximativement entre 3,72 V et 3,73 V, $\overline{FDC}$ ne passe au niveau BAS qu'à l'instant du palier 3,73 V. La gamme exacte des valeurs de V_A est:

$$3{,}72\ \text{V} - V_T \text{ à } 3{,}73\ \text{V} - V_T$$

Comme V_T est très petit, nous sommes justifié d'affirmer que cette gamme est approximativement entre 3,72 et 3,73 V, soit 10 mV la résolution du CNA. La figure 10-13 illustre tout ceci.

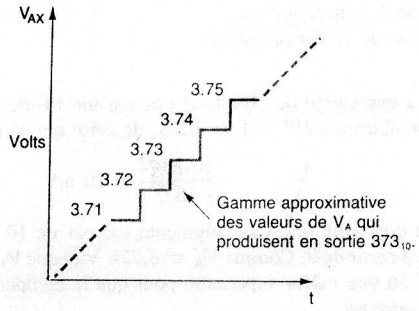


FIGURE 10-13 Exemple 10-13B.

Résolution et précision d'un CAN Comme on le soulignait à l'exemple 10-13A, la résolution d'un convertisseur AN est égale à la résolution du convertisseur NA qui se trouve dans son circuit. La tension de sortie du CNA, V_{AX}, est une forme d'onde en escalier qui progresse par pas discret jusqu'à ce que V_A soit dépassé. Donc V_{AX} est une approximation de V_A et le mieux auquel on puisse s'attendre est que V_{AX} soit à 10 mV près de V_A si la résolution (pas de progression) est 10 mV. Il est possible de traiter la résolution comme une erreur inhérente que l'on appelle souvent **erreur de quantification**. Cette erreur de quantification, que l'on peut diminuer en augmentant le nombre de bits du compteur et du convertisseur NA, est souvent spécifiée comme une erreur de une fois le poids du bit le plus faible. La valeur mesurée peut s'écarter de la valeur réelle d'une quantité égale à cette erreur de quantification en raison du pas de progression non nul. Nous verrons au problème 10-28 comment il est possible de modifier l'erreur de quantification pour qu'elle soit égale à la moitié du poids du bit le plus faible, ce qui correspond à la situation la plus courante.

Examinons cette question d'un point de vue différent: l'entrée V_A peut prendre une *infinité* de valeurs entre 0 V et la pleine échelle. Par contre, l'approximation V_{AX}, elle, ne peut occuper qu'un nombre fini de valeurs discrètes. Cela veut donc dire qu'il y aura une petite plage de valeurs V_A qui auront la même représentation numérique. À titre d'illustration, dans l'exemple 10-13B, toutes les valeurs de V_A comprises entre 3,720 V et 3,7299 V sont identifiées au moyen de 373 pas, et ont donc toutes la même représentation numérique. Autrement dit, V_A doit changer d'au moins 10 mV (la résolution) pour passer à la représentation numérique suivante.

Comme dans le convertisseur NA, la *précision* n'est pas reliée à la résolution, mais dépend de la précision des circuits constitutifs, comme le comparateur, les résistances de précision et les amplificateurs de niveau du CNA, les alimentations de référence etc. Une erreur fixée à 0,01 % PE indique que le résultat du convertisseur AN peut s'écarter de 0,01 % PE de la valeur idéale en raison de composants non idéaux. Cette erreur *s'ajoute* à l'erreur de quantification due à la résolution. Ces deux erreurs sont généralement du même ordre de grandeur pour un CAN donné.

EXEMPLE 10-14

Un certain CAN de 8 bits a une entrée pleine échelle de 2,55 V (c'est-à-dire quand V_A = 2,55 V la sortie numérique est 11111111). L'erreur spécifiée est de 1 % PE. Déterminez l'écart maximal qui peut exister entre la sortie V_{AX} et l'entrée analogique.

Solution Le pas de progression est 2,55 V/(2^8 − 1), ce qui donne exactement 10 mV. Donc même si ce CAN n'a aucune incertitude, la sortie numérique peut s'écarter de 10 mV de la valeur analogique parce que V_{AX} ne change que par pas de 10 mV. Il s'agit de l'erreur de quantification. Comme l'erreur spécifiée est 1 % PE, cela donne 1 % × 2,55 V = 2,55 mV. La valeur V_{AX} peut donc varier d'au plus 2,55 mV à cause des imprécisions des composants. Donc l'erreur totale possible est au plus 10 mV + 2,55 mV = 12,55 mV.

Prenons l'exemple d'un signal analogique de 1,268 V. Si la sortie du CNA était parfaitement exacte, l'escalier s'arrêterait au 127e palier. Cependant, supposons que V_{AX} est décalé de −2 mV, de sorte qu'il vaut 1,268 au 127e pas. Cette valeur ne suffit pas à mettre un terme à la conversion qui s'arrêtera au 128e pas. Donc, la sortie numérique équivalente au signal analogique de 1,268 sera $10000000_2 = 128$, soit une erreur de 12 mV.

Durée de conversion, t_C La durée de conversion est définie à la figure 10-12 b) comme l'intervalle de temps entre la fin de l'impulsion DÉBUT et l'activation du signal $\overline{FDC}$. Le compteur se met en marche avec comme valeur de départ zéro; il progresse jusqu'à ce que V_{AX} dépasse V_A, c'est à ce moment que $\overline{FDC}$ effectue une transition vers le niveau BAS et met un terme à la conversion. Il est évident que la durée de conversion, t_C, est fonction de V_A. Pour une plus grande valeur, l'escalier doit monter plus haut avant de dépasser V_A.

La durée de conversion maximale est celle d'une tension V_A qui est juste inférieure à la pleine échelle, de sortie que V_{AX} doit passer au dernier palier pour activer le signal $\overline{FDC}$. Dans le cas d'un convertisseur de N bits, elle vaut:

$$t_C(\text{max}) = 2^N - 1 \text{ cycles d'horloge}$$

Par exemple, le CAN de la figure 10-13A aura une durée de conversion maximale de

$$t_C(\text{max}) = (2^{10} - 1) \times 1\ \mu\text{s} = 1023\ \mu\text{s}$$

Parfois, la durée de conversion moyenne est indiquée; il s'agit de la moitié de la durée de conversion maximale. Dans le cas des convertisseurs rampe numérique, cette durée est

$$t_C(\text{moyen}) = \frac{t_C(\text{max})}{2} \approx 2^{N-1} \text{ cycles d'horloge}$$

Le principal inconvénient de la méthode de la rampe numérique est la durée de conversion qui double pratiquement chaque fois qu'on ajoute un bit au compteur, de sorte qu'on augmente sa résolution au détriment de la durée de conversion. Ce CAN ne convient donc pas aux applications où ont lieu fréquemment des conversions AN de signaux analogiques qui varient rapidement. Pour les applications lentes, la simplicité relative du convertisseur rampe numérique l'avantage par rapport aux CAN rapides plus complexes.

QUESTIONS DE RÉVISION

1. Décrivez le fonctionnement de base du CAN rampe numérique.
2. Définissez « erreur de quantification ».
3. Dites pourquoi la durée de conversion augmente proportionnellement avec la tension d'entrée analogique.
4. *Vrai ou faux:* Toute chose égale d'ailleurs, un CAN rampe numérique de 10 bits a une meilleure résolution mais une durée de conversion plus longue qu'un CAN de 8 bits.
5. Donnez un avantage et un inconvénient du CAN rampe numérique.
6. Pour le convertisseur de l'exemple 10-13A, trouvez la sortie numérique correspondant à $V_A = 1{,}345$. Même question pour $V_A = 1{,}342$ V.

10-9 ACQUISITION DES DONNÉES

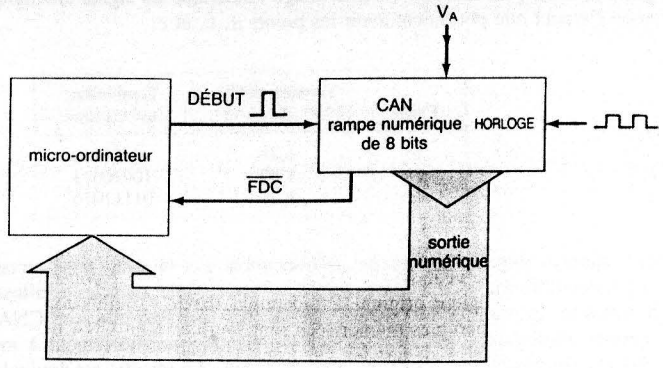
Les applications dans lesquelles des données analogiques doivent être *numérisées* (converties en un chiffre) et transférées dans une mémoire d'ordinateur sont nombreuses. Le processus de saisie par l'ordinateur des données analogiques numérisées s'appelle *l'acquisition des données*. Selon l'application, l'ordinateur peut manipuler les données de différentes façons. Quand on utilise un ordinateur pour mémoriser une donnée analogique (oscilloscope numérique ou enregistrement audio-numérique), ce dernier stocke les données puis les transfère à un CNA à un moment ultérieur pour qu'il reproduise les données analogiques. Quand on utilise un ordinateur pour la régulation d'un procédé, l'ordinateur examine les données ou les transforme au moyen de calculs afin de déterminer les niveaux de régulation à générer.

La figure 10-14 a) nous fait voir un micro-ordinateur qu'on a relié à un CAN rampe numérique en vue d'acquérir des données. L'ordinateur produit les impulsions DÉBUT qui lancent chaque nouvelle conversion AN. Le signal $\overline{\text{FDC}}$ produit par les CAN est communiqué à l'ordinateur. Ce dernier surveille le signal $\overline{\text{FDC}}$ pour savoir quand la conversion AN courante prend fin; ensuite il transfère la donnée numérique de la sortie du CAN dans la mémoire.

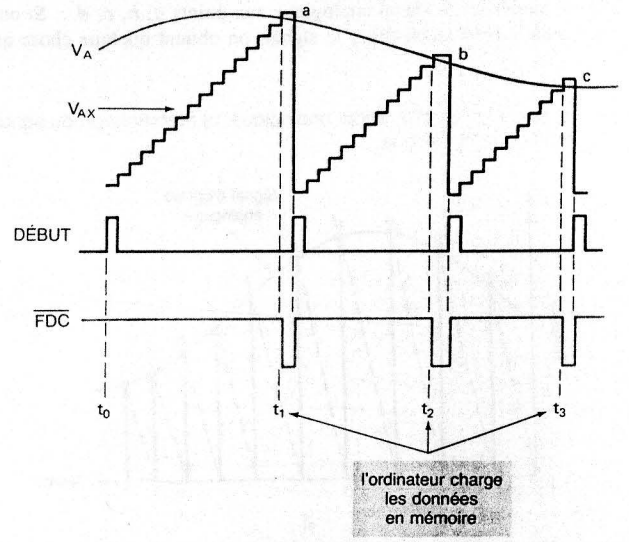
Les formes d'ondes de la figure 10-14 b) montrent comment l'ordinateur acquiert la version numérique d'un signal analogique, V_A. La forme d'onde en escalier V_{AX} que produit le CAN interne est superposée à la forme d'onde de V_A à des fins d'illustration. Le processus commence à l'instant t_0 quand l'ordinateur produit une impulsion DÉBUT qui lance un cycle de conversion AN. La conversion est achevée à l'instant t_1 quand le premier escalier dépasse V_A et que $\overline{\text{FDC}}$ passe au niveau BAS. Ce FD de la borne $\overline{\text{FDC}}$ signale à l'ordinateur que la sortie du CAN est une valeur numérique qui correspond à la valeur de V_A au point *a*, et que l'ordinateur peut charger cette donnée en mémoire.

L'ordinateur produit une nouvelle impulsion DÉBUT presque immédiatement après l'instant t_1 en vue de lancer un deuxième cycle de conversion. Notez que cela ramène l'escalier à 0 et que $\overline{\text{FDC}}$ retourne au niveau HAUT parce que l'impulsion DÉBUT réinitialise le compteur CAN. La deuxième conversion prend fin à l'instant t_2 quand l'onde en escalier dépasse de nouveau V_A. L'ordinateur charge en mémoire la valeur numérique correspondant au point *b*. Les mêmes étapes se répètent en t_3, en t_4 et ainsi de suite.

Le processus de lancement d'une impulsion DÉBUT, d'examen de la borne $\overline{\text{FDC}}$, et de mémorisation du résultat numérique du CAN est dirigé par un programme qu'exécute l'ordinateur. C'est ce programme d'acquisition des données qui fixe le nombre de points du signal analogique qui sera placé en mémoire.



a)



b)

FIGURE 10-14 a) Système type d'acquisition de données par un ordinateur; b) formes d'ondes illustrant la façon dont l'ordinateur lance chaque nouveau cycle de conversion pour ensuite charger la donnée numérique en mémoire à la fin de la conversion.

Reconstitution d'un signal numérisé La figure 10-14 b) contient un CAN qui fonctionne à sa vitesse maximale, puisqu'un nouveau signal DÉBUT est généré immédiatement après l'acquisition par l'ordinateur de la dernière valeur convertie. Remarquez que les durées de conversion ne sont pas égales, puisque la valeur du signal analogique d'entrée change. L'ordinateur mémorise les données numériques résultant des différentes conversions; il

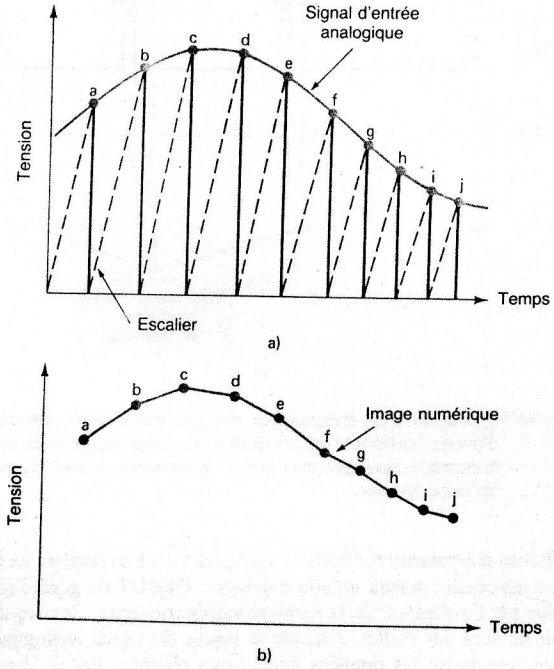
possède donc dans sa mémoire une image numérique du signal analogique. Par exemple, voici l'aspect que pourraient avoir les points *a*, *b*, et *c*:

Point	Tension réelle (V)	Équivalent numérique
a	1,74	10101110
b	1,47	10010011
c	1,22	01111010

Ces données numériques servent fréquemment par la suite pour reconstituer un signal approximatif du signal analogique original. Par exemple, dans les oscilloscopes numériques à mémoire, les valeurs numériques mémorisées sont fournies à un CNA qui redonne des tensions analogiques servant à dévier les électrons verticalement à mesure qu'ils sont déplacés horizontalement par une base de temps. Le résultat est équivalent à reconstruire l'original en joignant par des droites (vecteurs) tous les points numérisés. C'est ce qu'illustre la figure 10-15.

Sur la figure 10-15 a), nous pouvons voir un CAN qui effectue sans arrêt des conversions afin de numériser le signal analogique aux points *a*, *b*, *c*, *d*... Si on exploite les valeurs en ces points pour reconstituer le signal, on obtient quelque chose qui ressemble

FIGURE 10-15 a) Numérisation d'un signal analogique; b) reconstitution du signal à partir de son image numérique.



à la figure 10-15 b). Nous pouvons constater qu'il s'agit d'une reproduction assez fidèle de l'original. Cela s'explique par le fait que le signal analogique ne change pas radicalement entre chaque point numérisé. Si le signal analogique comprenait des composantes de haute fréquence, le CAN ne serait pas capable de suivre les variations et l'image reconstituée ne serait pas un reflet fidèle de l'original. C'est pour cette raison qu'il est important que la durée de conversion du CAN soit la plus courte possible afin que le signal analogique n'ait pas le temps de changer de façon significative entre chaque conversion. On comprend mieux à présent la nécessité des CAN dont les durées de conversion sont plus courtes que celles des simples convertisseurs rampe numérique. Nous allons examiner d'autres solutions dans les sections à venir.

QUESTIONS DE RÉVISION

1. Définissez le terme « numérisation d'un signal ».
2. Décrivez brièvement le processus d'acquisition des données par un ordinateur.

10-10 CAN PAR APPROXIMATIONS SUCCESSIVES (À PESÉES SUCCESSIVES)

Ce type de CAN est l'un de ceux qu'on retrouve le plus fréquemment. Il possède des circuits plus complexes que le CAN rampe numérique, mais son temps de conversion est beaucoup plus court. En outre, les convertisseurs par approximations successives (CAS) ont une durée de conversion fixe qui ne dépend pas de la valeur de l'entrée analogique.

Le montage de base de ce convertisseur est semblable à celui du convertisseur rampe numérique (voir figure 10-16 a)). Toutefois, ce convertisseur n'utilise pas de compteur pour alimenter l'entrée du convertisseur NA mais plutôt un registre. La logique de contrôle modifie le contenu du registre bit par bit jusqu'à ce que la donnée qui s'y trouve soit l'équivalent numérique du signal analogique V_A (compte tenu de la résolution du convertisseur). Le déroulement de ce processus est décrit au moyen d'un organigramme à la figure 10-16 b). Nous allons nous référer à cet organigramme pendant notre étude détaillée de l'exemple de la figure 10-17.

Pour les besoins de cet exemple, nous avons retenu un convertisseur 4 bits élémentaire dont le pas de progression est 1 V. Même si la plupart des CAS possèdent plus de bits et une précision plus fine que ces paramètres, leur principe de fonctionnement est identique à ce que nous allons présenter. À ce moment-ci, vous devriez n'avoir aucune difficulté à déduire que les quatre registres alimentant le CNA ont respectivement comme poids 8, 4, 2 et 1 V.

Supposons que l'entrée analogique soit $V_A = 10{,}4$ V. La première action est une mise à zéro par la logique de contrôle de tous les bits des registres; donc, $Q_3 = Q_2 = Q_1 = Q_0 = 0$. Nous convenons d'exprimer cette situation par la notation $[Q] = 0000$. La sortie du CNA est alors pour le moment $V_{AX} = 0$, comme l'illustre le schéma de la figure 10-17 pour la valeur t_0. Comme $V_{AX} < V_A$, la sortie du comparateur est HAUTE.

À l'instant suivant (t_1), la logique de contrôle met à 1 le bit du registre de poids fort et $[Q] = 1000$. Cela rend $V_{AX} = 8$ V, l'inégalité $V_{AX} < V_A$ est conservée et la sortie COMP reste HAUTE. Ce niveau HAUT est un signal à la logique de contrôle que la mise à 1 du bit de poids fort n'a pas suffit à amener V_{AX} au delà de V_A. Le résultat est de garder le bit de poids fort à 1.

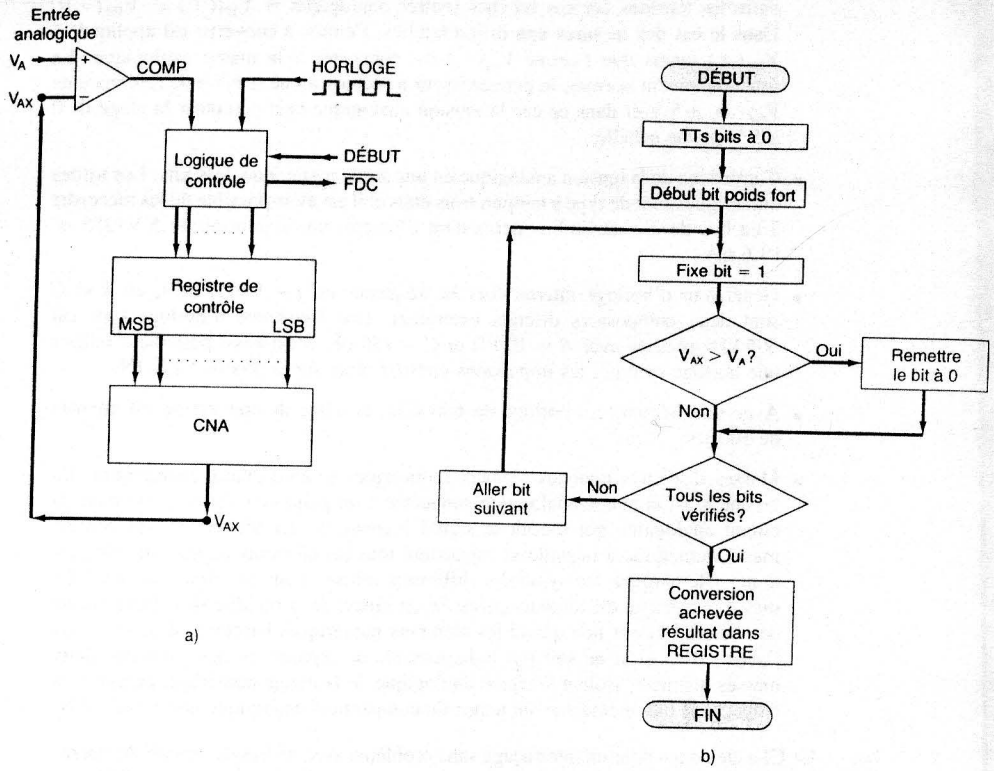
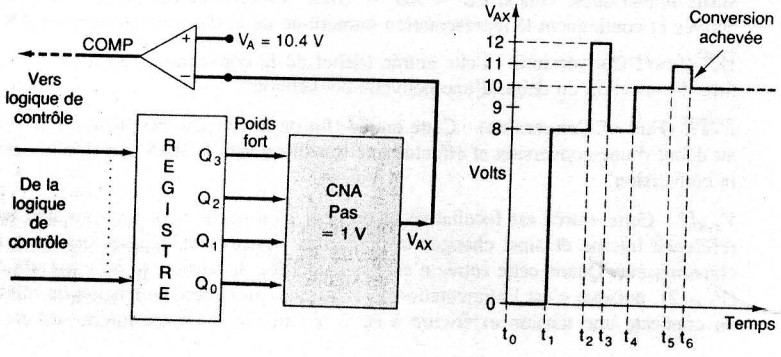


FIGURE 10-16 CNA par approximations successives: a) bloc fonctionnel simplifié b) organigramme.

FIGURE 10-17 Illustration du fonctionnement d'un CAS de 4 bits utilisant un CNA de 4 bits dont le pas de progression est 1 V.



La logique de contrôle se rend à présent au second bit, Q_2, qu'elle met à 1 afin de produire $[Q] = 1100$. Cette fois-ci à l'instant t_2, $V_{AX} = 12$ V, ce qui crée l'inégalité $V_{AX} > V_A$ et fait passer le signal COMP au niveau BAS. Ce BAS signale à la logique de contrôle que la valeur de V_{AX} est trop grande et sa réaction est de remettre ce bit à 0, à l'instant t_3. La valeur de V_{AX} revient à 8 V.

L'étape suivante, en t_4, est de changer le bit suivant Q_1 afin de former $[Q] = 1010$ et d'avoir $V_{AX} = 10$ V. Pour cette valeur, $V_{AX} < V_A$, le signal COMP est HAUT et la logique reçoit l'ordre de laisser le bit de Q_1 à 1.

La dernière étape consiste à mettre à 1 le dernier bit, Q_0. Cela donne le registre $[Q] = 1011$ et la tension numérique $V_{AX} = 11$ V. Comme à présent $V_{AX} > V_A$, le signal COMP passe au niveau BAS indiquant que la sortie numérique dépasse le signal analogique; la logique de contrôle reçoit l'ordre de remettre ce bit à 0, à l'instant t_6.

Comme à ce moment-ci tous les bits du registre ont été traités, la conversion est terminée et la logique de contrôle active le signal $\overline{FDC}$ qui indique que l'équivalent numérique trouvé pour le signal analogique se trouve dans le registre. Dans le cas de notre exemple, la sortie numérique correspondant à $V_A = 10{,}4$ V est $[Q] = 1010$, soit 10 V. On voit bien que cette valeur est *inférieure* à la valeur analogique; cette réponse est caractéristique de la méthode par approximations successives. Rappelons que dans la méthode à rampe numérique la sortie numérique est toujours équivalente à la tension du palier immédiatement au-dessus de V_A.

EXEMPLE 10-15

Un CAS de 8 bits possède une résolution de 20 mV. Quelle sera la sortie numérique correspondant à une entrée analogique de 2,17 V?

Solution

$$2{,}17 \text{ V}/20 \text{ mV} = 108{,}5$$

donc à la 108e étape le convertisseur produit $V_{AX} = 2{,}16$ V et à la 109e, 2,18 V. Le CAS fournit toujours comme réponse une valeur numérique correspondant au pas immédiatement *sous* V_A. Par conséquent, pour l'entrée $V_A = 2{,}17$ V, le convertisseur sort l'équivalent numérique $108_{10} = 01101100_2$.

Durée de la conversion Dans le fonctionnement que l'on vient de décrire, la logique de contrôle met successivement chaque bit à 1 et décide si elle conserve ou non le 1. Le traitement de chaque bit exige un cycle d'horloge, ce qui se traduit par une durée de conversion totale de N cycles d'horloge pour un CAS de N bits. Ou exprimé différemment:

$$t_C \text{ d'un CAS} = N \times 1 \text{ cycle d'horloge}$$

La durée de conversion est toujours la même *quelle que soit la valeur de* V_A. Cela s'explique par le fait que la logique de contrôle doit traiter chaque bit, qu'il soit mis à 1 ou non.

EXEMPLE 10-16

Comparez les durées de conversion maximales d'un CAN rampe numérique de 10 bits et d'un CAN par approximations successives de 10 bits; la fréquence du signal dl'horloge est de 500 KHz dans les deux cas.

Solution Pour le convertisseur rampe numérique, la durée de conversion maximale est donnée par:

$$(2^N - 1) \times (1 \text{ cycle d'horloge}) = 1023 \times 2\ \mu\text{s} = 2046\ \mu\text{s}$$

Pour le convertisseur par approximations successives, la durée de conversion est toujours 10 cycles d'horloge, soit:

$$10 \times 2\ \mu s = 20\ \mu s$$

Donc, ce dernier est presque 100 fois plus rapide que le convertisseur à rampe numérique.

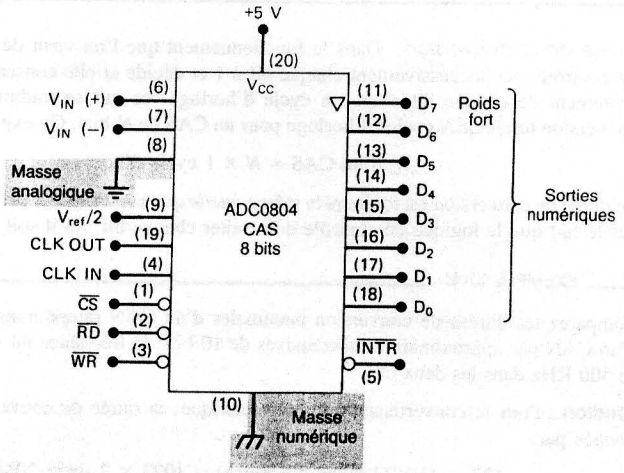
Comme les CAS ont des durées de conversion relativement courtes, on les retrouve dans les applications d'acquisition des données, puisque l'on peut avec eux recueillir plus de données dans un certain intervalle de temps. C'est une considération importante quand les signaux analogiques changent à une cadence assez rapide.

Parce que les CAS sont vendus sous forme intégrée, il est rarement nécessaire de concevoir les circuits de la logique de contrôle et nous n'en parlerons pas ici. Ceux qui seraient intéressés par ce sujet trouveront dans les manuels techniques des fabricants beaucoup de détails.

Un CI réel: le CAN par approximations successives ADC0804 Un grand nombre de fabricants de CI offrent des CAN ayant une grande diversité de plages de fonctionnement et de fonctions. Nous nous proposons d'examiner un de ces éléments afin de se faire une idée d'utilisation pratique dans les applications. La figure 10-18 reproduit le brochage du ADC0804, un boîtier intégré CMOS de 20 broches qui effectue la conversion AN au moyen de la méthode par approximations successives. Voici certaines de ses caractéristiques les plus importantes:

- Deux entrées analogiques: $V_{IN}(+)$ et $V_{IN}(-)$ pour permettre une entrée **différentielle**. On veut dire par là que l'entrée analogique convertie est la différence

FIGURE 10-18 L'ADC0804, un CAN 8 bits par approximations successives doté de sorties trois états. Les numéros entre parenthèses correspondent aux numéros des broches du boîtier.



entre les tensions sur ces broches [entrée analogique = $V_{IN}(+) - V_{IN}(-)$]. Dans le cas des mesures non différentielles, l'entrée à convertir est appliquée à $V_{IN}(+)$ tandis que l'entrée $V_{in}(-)$ est connectée à la masse analogique. En fonctionnement normal, le convertisseur a recours à une tension de référence de $V_{CC} = +5$ V et dans ce cas la tension analogique peut parcourir la plage de 0 à 5 V pleine échelle.

- Conversion de la tension analogique en une sortie numérique de 8 bits. Les sorties numériques sont de type à tampon trois états et il est alors possible de les raccorder à un bus de donnée. 8 bits permettent d'obtenir une résolution de 5 V/255 = 19,6 mV.
- Générateur d'horloge interne dont la fréquence est $f = 1/(1{,}1\ RC)$, où R et C sont deux composants discrets extérieurs. Une fréquence d'horloge type est 606 kHz réalisée avec $R = 10\ k\Omega$ et $C = 150$ pF. Il est aussi possible d'utiliser une horloge externe; les impulsions arrivent alors sur la broche CLK IN.
- Avec une fréquence d'horloge de 606 kHz, la durée de conversion est environ de 100 μs.
- Masses distinctes pour les signaux numériques et les signaux analogiques. La broche 8 est la masse analogique connectée à un point de référence commun du circuit analogique qui fournit le signal à convertir. La broche 10 agit comme masse numérique à laquelle se rapportent tous les éléments numériques du système. (Remarquez les symboles différents utilisés pour ces deux masses.) La masse numérique est toujours parasitée en raison de la rapidité des changements de courant qui ont lieu quand les éléments numériques basculent d'un état vers l'autre. Bien qu'il ne soit pas indispensable de séparer ces deux masses, deux masses distinctes isolent le signal analogique de la masse numérique parasitée et empêche le basculement avant temps du comparateur analogique interne au CAN.

Ce CI a été conçu pour un interfaçage sans problème avec un bus de donnée de micro-ordinateur. Pour cette raison, les désignations des entrées et des sorties du ADC0804 sont calquées sur celles des fonctions équivalentes dans les systèmes à micro-ordinateur. Voici la définition du rôle de ces entrées et de ces sorties:

$\overline{CS}$ (Chip Select) Cette entrée (sélection de la puce) doit être à son niveau vrai (BAS) pour que les entrées $\overline{RD}$ et $\overline{WR}$ agissent. Quand $\overline{CS}$ est au niveau HAUT, les sorties numériques sont dans leur état de haute impédance et aucune conversion ne se déroule.

$\overline{RD}$ (Output Enable) Cette entrée (validation de la sortie) sert à valider les tampons de sortie numériques. Quand $\overline{CS} = \overline{RD}$ = BAS, les broches des sorties numériques sont actives et contiennent la représentation numérique de la *dernière* conversion AN.

$\overline{WR}$ (Start Conversion) Cette entrée (début de la conversion) reçoit un signal BAS à titre d'indication du début d'une nouvelle conversion.

$\overline{INTR}$ (End of Conversion) Cette entrée (fin de la conversion) passe au niveau HAUT au début d'une conversion et effectue une transition vers le BAS afin d'indiquer la fin de la conversion.

$V_{ref}/2$ Cette entrée est facultative et on peut s'en servir pour imposer une tension de référence interne et ainsi changer la plage des signaux analogiques que peut traiter le convertisseur. Quand cette entrée n'est pas raccordée, la tension qu'on y mesure est 2,5 V ($V_{CC}/2$), puisque c'est l'alimentation V_{CC} qui agit alors comme tension de référence. Si on connecte une tension extérieure à cette broche, la référence interne est modifiée et

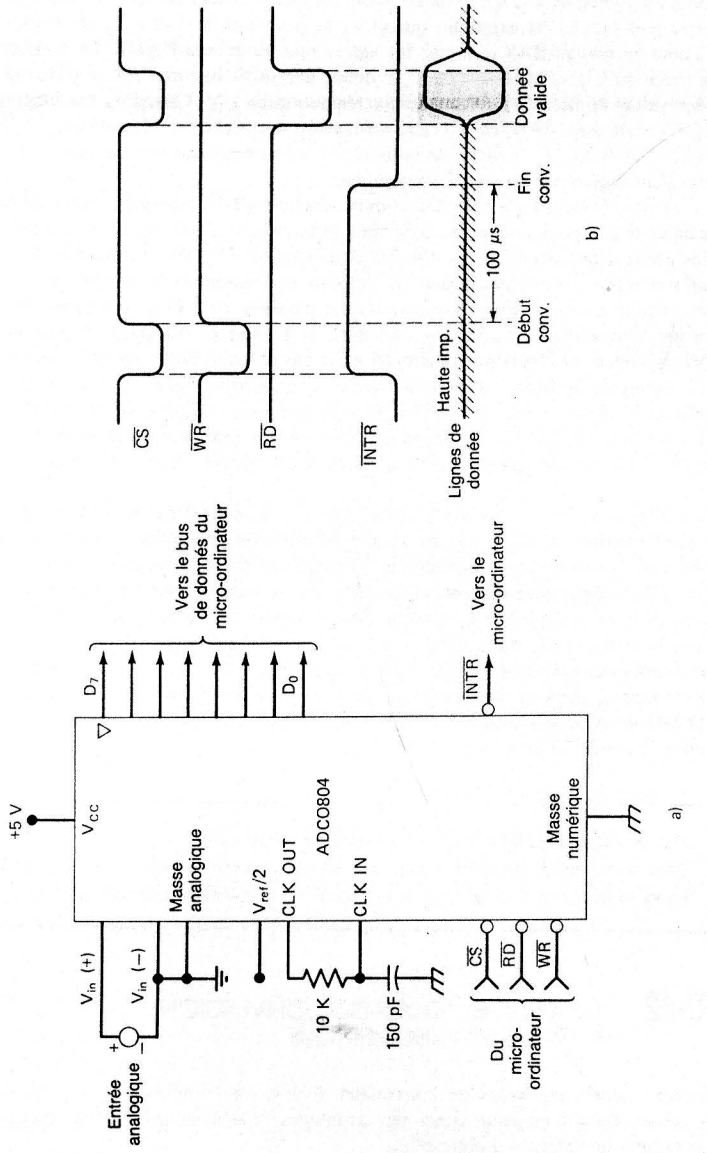


FIGURE 10-19 a) Connexions habituelles durant l'interfaçage d'un ADC0804 avec un micro-ordinateur; b) signaux types durant un processus d'acquisition de données.

devient deux fois la tension fournie de l'extérieur; la plage des tensions analogiques change de façon correspondante. Voici un tableau illustrant ceci:

$V_{ref}/2$	Plage des entrées analogiques (V)	Résolution (mV)
Ouverte	0–5	19,6
2,25	0–4,5	17,6
2,0	0–4	15,7
1,5	0–3	11,8

CLK OUT Cette entrée (sortie horloge) est connectée à une résistance quand on utilise l'horloge interne. C'est sur cette broche que l'on relève le signal d'horloge.

CLK IN Cette entrée (entrée horloge) sert à recevoir les signaux d'une horloge externe ou à connecter un condensateur quand on utilise l'horloge interne.

La figure 10-19 a) montre les connexions habituelles durant l'interfaçage d'un ADC0804 avec un micro-ordinateur dans une application d'acquisition de données. C'est le micro-ordinateur qui contrôle la conversion en envoyant les signaux $\overline{CS}$ et $\overline{WR}$. Ensuite, il prélève la donnée de sortie du CAN en générant les signaux $\overline{CS}$ et $\overline{RD}$ dès qu'il a décelé un front descendant sur la borne $\overline{INTR}$, signal de la fin de la conversion. Les formes d'ondes de la figure 10-19 b) illustrent le séquencement des signaux durant le processus d'acquisition. Notez que $\overline{INTR}$ passe au niveau HAUT au moment ou $\overline{CS}$ et $\overline{WR}$ passent au niveau BAS, mais que la conversion proprement dite ne débute pas tant que $\overline{CS}$ et $\overline{WR}$ ne sont pas revenus à 1. Observez en outre que les lignes de donnée de sortie du CAN restent dans leur état haute impédance jusqu'au moment où le micro-ordinateur active $\overline{CS}$ et $\overline{RD}$; c'est à cet instant que les tampons de donnée du CAN sont validés et envoient la donnée numérique du CAN sur le bus de donnée. Les lignes de donnée retournent dans leur état haute impédance quand $\overline{CS}$ et $\overline{RD}$ sont remis dans l'état HAUT.

QUESTIONS DE RÉVISION

1. Quel est le principal avantage d'un CAS par rapport à un CAN rampe numérique?
2. Quel est son principal inconvénient par rapport à un CAN rampe numérique?
3. *Vrai ou faux:* La durée de conversion d'un CAS augmente avec la tension analogique à convertir.
4. Répondez aux questions suivantes concernant le ADC0804.
   a) Quelle est sa résolution en bits?
   b) Quelle est la plage normale des tensions d'entrée?
   c) Décrivez le rôle des entrées $\overline{CS}$, $\overline{WR}$, et $\overline{RD}$.
   d) Quel est le rôle de la sortie $\overline{INTR}$?
   e) Pourquoi possède-t-il deux masses distinctes?

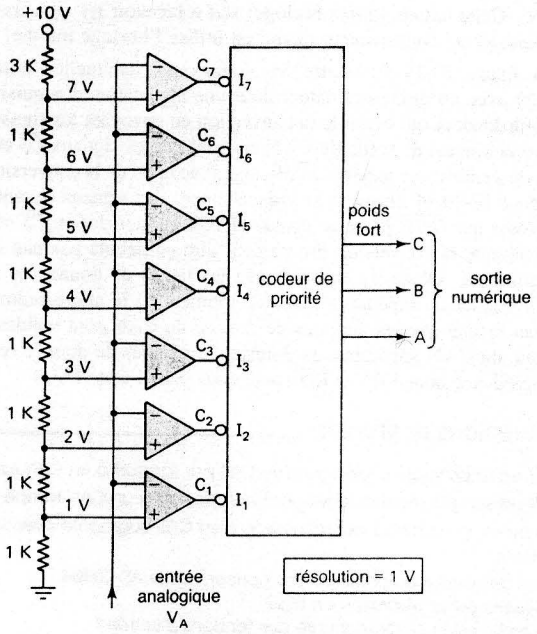
10-11 CAN PARALLÈLES (FLASH)

Il s'agit ici du CAN le plus rapide qui soit, et également de celui qui contient le plus grand nombre de circuits. Par exemple, un CAN parallèle de 6 bits exige 63 comparateurs analogiques, une unité de 8 bits en exige 255 et un convertisseur de 10 bits en exige 1023. Le grand nombre de comparateurs nécessaires limite dans la pratique les convertisseurs parallèles *discrets*. Par contre, les convertisseurs parallèles CI atteignent couramment 8 bits, et plusieurs fabricants annoncent pour prochainement des unités de 9 et de 10 bits.

Nous décrivons ici le principe de fonctionnement d'un convertisseur parallèle de 3 bits seulement afin de ne pas trop compliquer les choses. Lorsque vous comprendrez le fonctionnement d'un convertisseur de 3 bits, vous pourrez sans difficulté appliquer son principe à des convertisseurs travaillant avec un plus grand nombre de bits.

Le convertisseur parallèle de la figure 10-20 a) a une résolution de 3 bits et un pas de progression de 1 V. Le diviseur de tension établit des niveaux de référence pour chaque comparateur de sorte qu'on retrouve sept niveaux dont les valeurs sont 1 V (poids le plus faible), 2 V, 3 V, ..., et 7 V (pleine échelle). L'entrée analogique V_A est appliquée à l'autre entrée de chaque comparateur.

FIGURE 10-20 a) CAN parallèle de 3 bits; b) sa table de vérité.



a)

Entrée analogique	Sorties du comparateur							Sortie numérique		
V_A	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C	B	A
< 1 V	1	1	1	1	1	1	1	0	0	0
> 1 V, < 2 V	0	1	1	1	1	1	1	0	0	1
> 2 V, < 3 V	0	0	1	1	1	1	1	0	1	0
> 3 V, < 4 V	0	0	0	1	1	1	1	0	1	1
> 4 V, < 5 V	0	0	0	0	1	1	1	1	0	0
> 5 V, < 6 V	0	0	0	0	0	1	1	1	0	1
> 6 V, < 7 V	0	0	0	0	0	0	1	1	1	0
> 7 V	0	0	0	0	0	0	0	1	1	1

b)

Si $V_A < 1$ V, toutes les sorties des comparateurs C_1–C_7 sont au niveau HAUT. Si $V_A > 1$ V, une au moins des sorties des comparateurs est au niveau BAS. Les sorties des comparateurs sont délivrées à un codeur de priorité vrai au niveau BAS qui produit une sortie binaire égale à la sortie du comparateur qui se trouve au niveau BAS dont le nombre est le plus élevé. Par exemple, quand V_A se trouve entre 3 et 4 V, les sorties C_1, C_2 et C_3 sont au niveau BAS et toutes les autres sont au niveau HAUT. Le codeur de priorité ne réagit qu'à la valeur basse de C_3 et donne une sortie binaire $CBA = 011$, qui représente l'équivalent numérique de V_A avec une résolution de 1 V. Quand V_A est supérieure à 7 V, C_1–C_7 sont tous BAS, et la sortie du codeur sera $CBA = 111$, ce qui est l'équivalent numérique de V_A. Le tableau de la figure 10-20 b) nous fait voir les réponses pour toutes les valeurs possibles du signal analogique.

Le CAN de la figure 10-20 a comme résolution 1 V, puisque l'entrée analogique doit changer de 1 V pour amener la sortie numérique au pas suivant. Pour obtenir des résolutions plus fines, il faut accroître les niveaux de la tension d'entrée (c'est-à-dire utiliser un plus grand nombre de résistances dans le diviseur de tension) et le nombre de comparateurs. Par exemple, pour réaliser un convertisseur parallèle de 8 bits, le nombre de niveaux de tension doit être $2^8 = 256$, en comptant le niveau 0. Et pour obtenir cela, il faut 256 résistances et 255 comparateurs (il n'y a pas de comparateur pour le niveau 0 V). Les 255 sorties de comparateur sont délivrées à un circuit codeur de priorité qui fournit en sortie un code de 8 bits correspondant à la sortie du comparateur de rang le plus élevé dont la sortie est BASSE. De façon générale, un convertisseur parallèle de N bits a besoin de $2^N - 1$ comparateurs, de 2^N résistances et des circuits de la logique de codage.

Durée de conversion Il n'y a pas de signal d'horloge dans un convertisseur parallèle, et c'est pourquoi il n'y a pas de synchronisation ou de séquencement à considérer. La conversion se produit continuellement. Quand la valeur de l'entrée analogique change, les sorties du comparateur suivent, ce qui entraîne la modification des sorties du codeur. La durée de conversion est le temps qui s'écoule entre un changement de V_A et l'apparition d'une nouvelle sortie numérique; cette durée est fonction seulement des retards de propagation des comparateurs et des circuits logiques de codage. C'est donc dire que la durée de conversion dans un convertisseur parallèle est extrêmement courte. Par exemple, le AD9002 de Analog Devices est un CAN parallèle de 8 bits dont la durée de conversion nominale est de 10 ns.

QUESTIONS DE RÉVISION

1. *Vrai ou faux:* Un CAN parallèle ne contient pas de CNA.
2. Dites combien il y aurait de comparateurs dans un convertisseur parallèle de 12 bits.
3. Énoncez le principal avantage et le principal inconvénient d'un convertisseur parallèle.

10-12 AUTRES MÉTHODES DE CONVERSION ANALOGIQUE-NUMÉRIQUE

Certaines autres méthodes de conversion analogique-numérique sont exploitées depuis un certain temps, chacune ayant des avantages et des inconvénients. Voici une brève description de certaines d'entre elles.

CAN rampe numérique bidirectionnel (CAN de poursuite) Nous avons vu que le CAN rampe numérique fonctionnait relativement lentement en raison de la nécessité de remettre

le compteur à zéro au début de chaque conversion. L'escalier débute toujours à zéro et se construit en montant jusqu'au point de « basculement » où V_{AX} excède V_A et où la sortie du comparateur effectue une transition vers le niveau BAS. Le temps entre la remise à zéro de l'escalier et sa remontée vers un nouveau point de basculement est vraiment gaspillé. Le **CAN rampe numérique bidirectionnel** élimine ce temps inutile grâce à l'utilisateur d'un compteur/décompteur.

Le compteur/décompteur remplace le compteur qui alimente le CNA. Il est conçu pour compter progressivement tant que la sortie du comparateur indique que $V_{AX} < V_A$ et pour compter régressivement tant que la sortie du comparateur indique que $V_{AX} > V_A$. De cette façon, la sortie du CNA évolue toujours dans le même sens que la valeur V_A. Chaque fois que le comparateur bascule, c'est une indication que V_{AX} a rencontré la valeur V_A, l'équivalent numérique de V_A se trouve alors dans le compteur: la conversion est achevée.

Quand une nouvelle conversion est initiée, le compteur ne se *remet pas à zéro*; au contraire, il commence à compter progressivement ou régressivement, selon la sortie du comparateur, à partir de sa dernière valeur. Il compte ainsi jusqu'à ce que l'escalier rencontre le nouveau V_A, ce qui termine cette conversion. La forme d'onde de V_{AX} renferme alors des portions positives et des portions négatives et elle semble « poursuivre » le signal V_A. Pour cette raison, on désigne souvent cet élément sous le nom de **CAN de poursuite**.

Il découle clairement de tout cela que les durées de conversion sont le plus souvent inférieures dans cet élément, parce qu'on ne revient pas à zéro à chaque fois, mais plutôt que l'on compte à partir de la dernière valeur convertie. Évidemment, la durée t_C dépend de la valeur de V_A et n'est pas constante.

CAN tension-fréquence Le CNA tension-fréquence est d'une construction plus rudimentaire que les autres CNA, parce qu'on n'y retrouve pas de CAN. C'est plutôt un *oscillateur régularisé par une tension linéaire (VCO)* qui fournit une fréquence de sortie proportionnelle à la tension d'entrée. La tension analogique à convertir alimente le *VCO* en vue de générer une fréquence de sortie. Cette fréquence est ensuite délivrée à un compteur qui dénombre ses états pendant une certaine durée. Le nombre final se trouvant dans le compteur est l'équivalent numérique de la tension analogique.

À titre d'illustration, supposons un *VCO* qui génère une fréquence de 10 kHz pour chaque volt se trouvant sur ses bornes d'entrée (c.-à-d. 1 V produit 10 kHz, 1,5 V produit 15 kHz, 2,73 V produit 27,3 kHz). Si la tension d'entrée est 4,54 V, la sortie du *VCO* sera 45,4 kHz, un signal qui déclenche un compteur pendant, par exemple, 10 ms. Après avoir dénombrer pendant 10 ms, le compteur contient le nombre 454, soit la représentation numérique de 4,54 V.

Malgré sa simplicité, cette méthode n'est pas très précise à cause de la difficulté de concevoir un *VCO* ayant un degré de précision meilleur que 0,1 pour cent.

L'une des principales applications de ce convertisseur est son implantation dans des milieux industriels très parasités où les signaux analogiques doivent être transmis du transducteur à l'ordinateur. Les faibles signaux analogiques peuvent être grandement perturbés par le bruit s'ils sont transmis directement. Une meilleur solution est donc de délivrer le signal analogique à un *VCO* qui se chargera de produire un signal numérique dont la fréquence de sortie suit les variations de l'entrée analogique. Ce signal numérique est communiqué à l'ordinateur sans subir de détérioration par le bruit. Les circuits de l'ordinateur de régulation comptent les impulsions numériques (c.-à-d. réalisent la fonction de comptage de la fréquence) et donne en sortie une valeur numérique qui est l'équivalent de l'entrée analogique initiale.

CAN double rampe Le convertisseur double rampe est l'un des éléments ayant la durée de conversion la plus longue (entre 10 et 100 ms); son principal avantage est son bas prix

puisqu'il ne contient pas de composants de précision comme un *VCO* ou un CNA. Le principe de base du convertisseur suppose la charge et la décharge *linéaire* d'un condensateur au moyen de courants constants. D'abord, le condensateur est chargé par un courant constant dérivé du signal analogique d'entrée V_A, pendant une durée donnée. À la fin de cette durée, le condensateur est déchargé linéairement au moyen d'un courant constant dérivé d'une référence de tension précise, $V_{réf}$. Quand la tension du condensateur atteint zéro, la décharge linéaire est achevée. Pendant la durée de la décharge, une référence de fréquence numérique alimente un compteur qui dénombre les impulsions. La durée de la décharge est proportionnelle à la tension initiale dans le condensateur. Ainsi, à la fin du temps de décharge, le compteur contient un nombre qui est la contrepartie numérique de la tension initiale du condensateur, elle-même proportionnelle à V_A.

Au faible coût du convertisseur double rampe, il faut ajouter un autre avantage: sa faible sensibilité au bruit et aux variations de la valeur des composants dues aux changements thermiques. En raison de sa durée de conversion très longue, on n'utilise jamais ce convertisseur dans les applications d'acquisition de données. La durée de conversion n'est toutefois pas une caractéristique essentielle dans certaines applications, comme les voltmètres ou multimètres numériques; c'est la raison pour laquelle on le retrouve dans ces types d'appareils.

QUESTIONS DE RÉVISION

1. Par quel aspect le CAN rampe numérique bidirectionnel est-il supérieur au CAN rampe numérique?
2. Nommez le principal élément d'un CAN tension-fréquence.
3. Nommez deux avantages et un inconvénient du CAN double rampe.
4. Nommez trois types de CAN qui ne font pas usage d'un CNA.

10-13 VOLTMÈTRE NUMÉRIQUE

Un voltmètre numérique convertit un signal analogique en sa représentation DCB qui est par la suite décodée et affichée. La figure 10-21 nous montre un voltmètre numérique de 3 chiffres dans lequel on retrouve un CAN rampe numérique (à l'intérieur du rectangle en lignes tiretées). Trois compteurs DCB en cascade fournissent les données d'entrée à un convertisseur DCB-analogique de 3 chiffres dont le pas de progression est 10 mV et la tension pleine échelle, 9,99 V. Chaque étage du compteur DCB attaque également un registre de 4 bits qui alimente un pilote/décodeur et un afficheur. Le contenu des compteurs DCB est transféré au registre à la fin de chaque cycle de conversion, de sorte que les afficheurs ne réagissent pas quand les compteurs progressent ou sont recyclés à zéro; ils affichent seulement le nombre final qui représente la tension inconnue.

Les impulsions d'horloge arrivent à l'entrée d'une porte qui s'ouvre selon la valeur de la sortie d'un comparateur. Tant que $V_A > V_{AX}$, COMP = 1 et le compteur reçoit les impulsions des tracés de la porte ET. À mesure que le compteur progresse, la forme d'onde de V_{AX} monte par pas de 10 mV jusqu'à ce que V_{AX} dépasse V_A (voir figure 10-22). À ce moment, la sortie du COMP passe à 0 et bloque la porte ET ce qui arrête le comptage des impulsions d'horloge. Le front descendant à la sortie du COMP déclenche également le MONOSTABLE 1 qui produit une impulsion en Q_1 de 1 μs. Le front montant en Q_1 devient le front montant d'horloge qui commande le transfert des sorties des compteurs DCB dans leur registre respectif. Le front descendant de Q_1 déclenche un deuxième monostable, qui produit aussi une impulsion de 1 μs pour réinitialiser tous les compteurs.

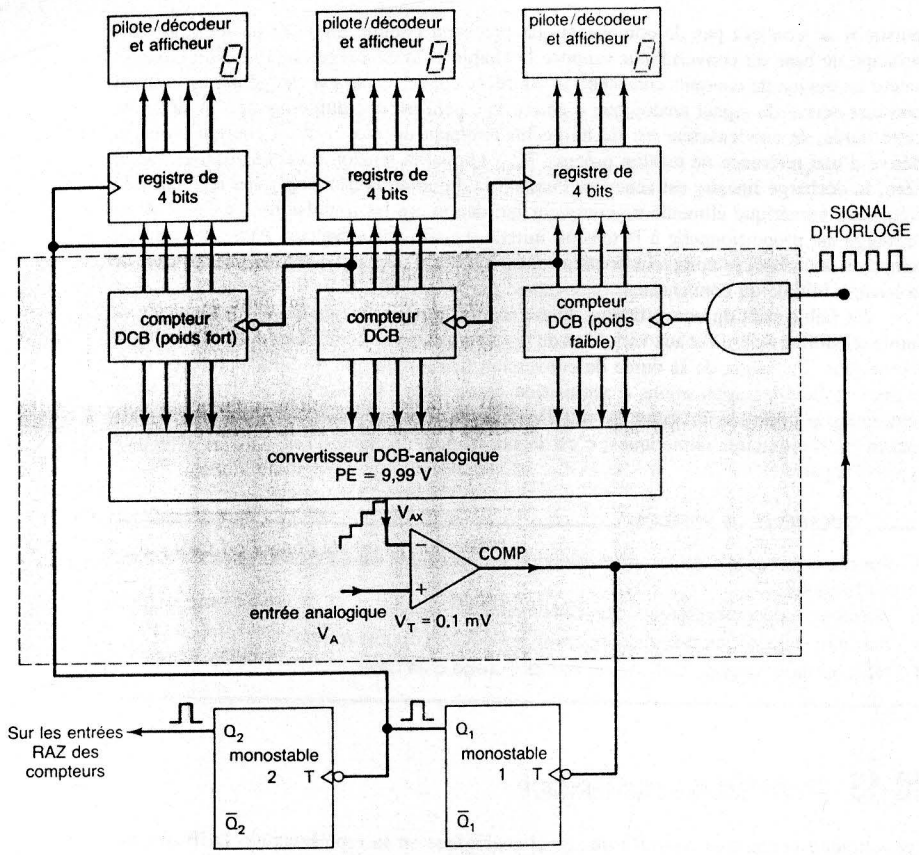


FIGURE 10-21 Voltmètre numérique à conversion ininterrompue au moyen d'un CAN rampe numérique.

Ceci ramène V_A à 0 V et le COMP au niveau HAUT, amorce le comptage des impulsions et initie un nouveau cycle de conversion.

Ainsi, ce voltmètre numérique effectue une conversion après l'autre. Évidemment, les registres mémoires empêchent les afficheurs de changer au rythme du processus de conversion. Les chiffres affichés ne changent que si V_A change, parce qu'un contenu différent des compteurs est transféré dans les registres à la fin du cycle de conversion.

Un exemple numérique devrait aider à fixer les idées sur le fonctionnement de ce circuit. Supposons que V_A soit 6,372 V. Pour que la sortie COMP bascule au niveau BAS, V_{AX} doit dépasser 6,3721 V. Étant donné que la sortie du CNA progresse par pas de 10 mV, il faut

$$\frac{6{,}3721\ \text{V}}{10\ \text{mV}} = 637{,}21 \longrightarrow 638\ \text{pas}$$

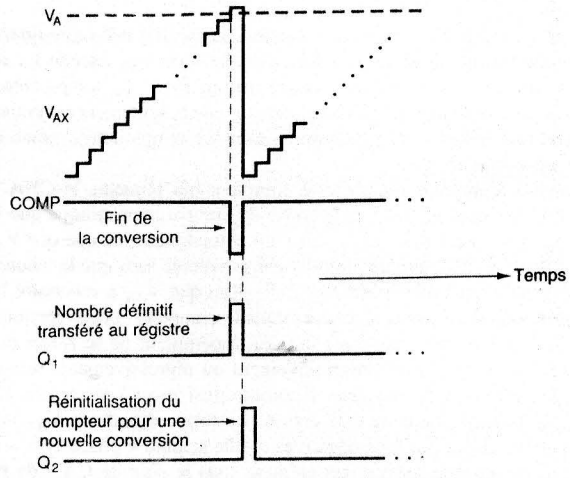


FIGURE 10-22 Formes d'ondes du voltmètre numérique.

Donc, le compteur doit dénombrer 638 cycles d'horloge qui seront transférés dans les registres et affichés par la suite. Une petite DEL peut être utilisée pour afficher un point décimal, de sorte que l'opérateur lira 6.38 V.

EXEMPLE 10-17

Expliquez ce qui se passe si V_A est supérieur à 9,99 V.

Solution Si $V_A > 9{,}99$ V, la sortie du comparateur reste à 1, et les impulsions d'horloge parviennent continuellement dans le compteur. Le compteur répétera sans arrêt le dénombrement jusqu'à 999 et sera recyclé à 000. Le monostable Q_1 ne sera *jamais* déclenché et les contenus des registres resteront toujours les mêmes. Un voltmètre numérique bien conçu est toujours doté d'un élément qui détecte une telle condition *hors calibre* et met en marche un certain type d'indicateur hors calibre. Une méthode possible consiste à ajouter un treizième bit à la chaîne des compteurs. Ce bit passe à 1 seulement si le compteur effectue la transition 999 à 000, et il signale une condition hors calibre.

Il est possible de modifier le voltmètre numérique étudié pour qu'il lise des tensions d'entrée dans plusieurs calibres, en intercalant un amplificateur ou un atténuateur convenable entre V_A et le comparateur. Par exemple, si V_A est 63,72 V, on peut l'atténuer d'un facteur de 10, de sorte que sur l'entrée + du comparateur la tension sera 6,372 V et le chiffre appliqué à la fin de la conversion sera 638. Le point décimal sera placé devant le chiffre de poids le plus faible, de sorte qu'on lira 63.8 V.

Le voltmètre numérique peut être transformé afin de servir en tant que multimètre numérique pour la mesure aussi bien du courant, de la résistance que de la tension. Pour mesurer un courant, il faut que l'intensité inconnue traverse une résistance de référence et développe ainsi une différence de potentiel. Pour mesurer une résistance, on force une référence de courant fixe dans la résistance à mesurer en vue de développer, là-aussi, une différence de potentiel. Dans chacun de ces cas, la tension obtenue est délivrée à un circuit

semblable à celui de la figure 10-21, qui la mesure et l'affiche. Il faudra ajouter à ce circuit un facteur d'atténuation d'échelle afin que la valeur affichée corresponde bien à la grandeur (courant ou résistance) mesurée.

Les tensions alternatives peuvent être déterminées par le même voltmètre si au préalable on les transforme en tensions continues. C'est ce qu'on fait avec un circuit précis de détection de crête. De nouveau, il faut prévoir un facteur d'échelle afin de transformer la valeur de crête en valeur efficace.

QUESTIONS DE RÉVISION

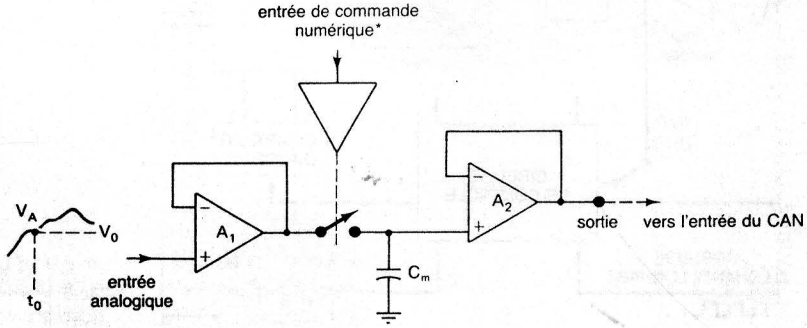
1. Quelle est la fonction des registres dans le circuit du voltmètre numérique?
2. Quelle est la fonction des deux monostables?

10-14 CIRCUITS ÉCHANTILLONNAGE ET MÉMORISATION

Quand on raccorde directement une tension analogique à l'entrée d'un CAN, le processus de conversion peut se trouver complètement perturbé si la tension analogique change pendant la durée de la conversion. La stabilité du processus de conversion s'améliore si on utilise un *circuit échantillonnage et mémorisation* pour garder constante la tension analogique pendant qu'a lieu la conversion AN. Un schéma simplifié d'un circuit échantillonnage et mémorisation (E/M) est reproduit à la figure 10-23.

Le circuit E/M contient un amplificateur tampon de gain un, A_1, qui offre une haute impédance au signal analogique et présente une basse impédance de sortie qui permet au condensateur de mémorisation, C_m, de se charger rapidement. Ce condensateur est raccordé à la sortie de A_1 quand l'interrupteur commandé numériquement est fermé. On parle dans ce cas du fonctionnement **échantillonnage**. L'interrupteur est fermé suffisamment longtemps pour que C_m se charge jusqu'à la tension courante du signal d'entrée. Par exemple, si l'interrupteur est fermé à l'instant t_0, la sortie A_1 charge rapidement C_m jusqu'à la tension V_0. Quand l'interrupteur est ouvert, C_m **mémorise** cette tension et la sortie de A_2 applique cette tension au CAN. L'amplificateur tampon de gain un, A_2, a une haute

FIGURE 10-23 Schéma simplifié d'un circuit échantillonnage et mémorisation.



*commande = 1 interrupteur fermé, mode échantillonnage
commande = 0 interrupteur ouvert, mode mémorisation

impédance d'entrée, de sorte que le condensateur ne se décharge presque pas pendant la durée de conversion du CAN. Ainsi l'entrée du convertisseur est pratiquement une tension d'entrée continue V_0.

Dans un système d'acquisition des données commandé par ordinateur, comme celui déjà discuté, l'interrupteur échantillonnage et mémorisation pourrait être commandé par un signal numérique envoyé par l'ordinateur. Le signal de l'ordinateur ferme l'interrupteur pour charger C_m jusqu'à une nouvelle valeur échantillonnée de la tension analogique; le temps pendant lequel l'interrupteur reste fermé est appelé le **temps d'acquisition** et il dépend de la valeur de C_m et des caractéristiques du circuit EM. Le LF198 est un circuit E/M intégré dont le temps d'acquisition typique est de 4 μs pour C_m = 1000 pF et de 20 μs pour C_m = 0,01 μF. Le signal de l'ordinateur ouvre ensuite l'interrupteur pour permettre à C_m de conserver sa valeur et de fournir une tension analogique relativement constante pour la sortie de *A2*. Par exemple, avec un LF198, la tension du condensateur se décharge généralement au rythme d'environ 30 mV par seconde dans le cas d'un condensateur de 1000 pF.

QUESTIONS DE RÉVISION

1. Décrivez le fonctionnement d'un circuit échantillonnage et mémorisation.
2. *Vrai ou faux:* Les amplificateurs du circuit EM ont un rôle d'amplification de la tension.

10-15 MULTIPLEXAGE

Quand il est nécessaire de convertir des signaux analogiques provenant de plusieurs sources, on peut recourir à une technique de multiplexage afin d'utiliser en temps partagé un convertisseur AN. Le schéma de base d'une telle solution est illustré à la figure 10-24 pour le cas d'un système d'acquisition à trois voies. Les interrupteurs S_1, S_2 et S_3 servent à aiguiller séquentiellement chacun des signaux analogiques vers le convertisseur AN. Les circuits de commande régissent le fonctionnement de ces interrupteurs, généralement des interrupteurs à semiconducteurs, afin qu'il n'y ait qu'un interrupteur de fermé à la fois. Les circuits de commande produisent également l'impulsion DÉBUT du CAN. Voici comment fonctionne un tel système:

1. Le circuit de commande ferme S_1, ce qui place V_{A1} sur l'entrée du CAN.
2. Une impulsion DÉBUT est générée et le CAN convertit V_{A1} en son équivalent numérique. S_1 demeure fermé tant que la conversion n'est pas complétée.
3. Les sorties du CAN représentant V_{A1} sont transférées dans un autre endroit. Dans de nombreux cas, cet endroit est la mémoire d'un ordinateur.
4. Le circuit de commande ouvre S_1 et ferme S_2 pour connecter V_{A2} à l'entrée du CNA.
5. Les étapes 2 et 3 sont répétées.
6. S_2 est ouvert et S_3 est fermé pour placer sur l'entrée du convertisseur V_{A3}.
7. Les étapes 2 et 3 sont répétées.

L'horloge de multiplexage contrôle le rythme avec lequel les signaux analogiques sont aiguillés séquentiellement vers le CNA. Le rythme maximal est déterminé par le retard des interrupteurs et la durée de conversion. Le retard de l'interrupteur peut être minimisé en utilisant des interrupteurs à semiconducteurs comme les interrupteurs bilatéraux CMOS

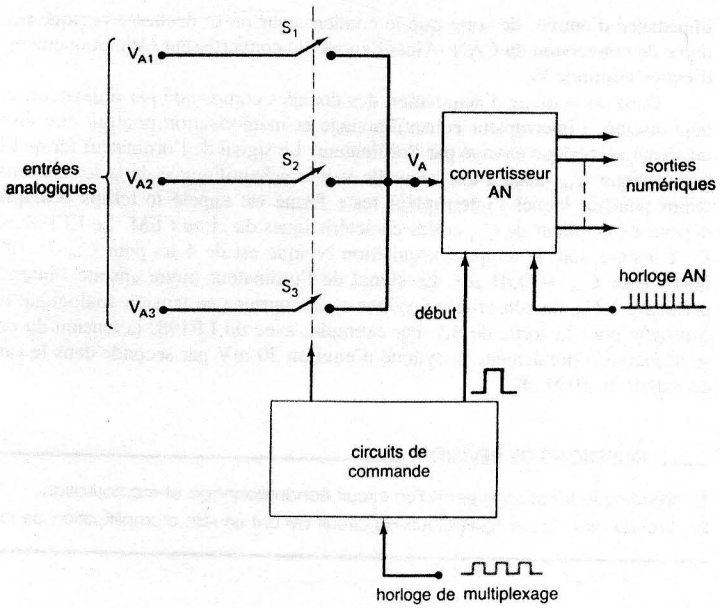


FIGURE 10-24 Conversion de plusieurs entrées analogiques par un seul convertisseur AN au moyen du multiplexage.

décrits au chapitre 8. Il pourra être nécessaire de placer un circuit échantillonnage et mémorisation à l'entrée du CAN si les entrées analogiques changent de façon assez importante pendant la durée de conversion du CAN.

De nombreux CAN intégrés possèdent dans leur boîtier le circuit de multiplexage. Le ADC0808, par exemple, peut multiplexer 8 entrées analogiques différentes vers un seul CAN. Pour ce faire, il utilise un code d'entrée de sélection de 3 bits qui établit quelle entrée analogique est raccordée au convertisseur.

10-16 OSCILLOSCOPE NUMÉRIQUE À MÉMOIRE (ONM)

Comme dernier exemple d'application des convertisseurs AN et NA, examinons le fonctionnement d'un oscilloscope numérique à mémoire. L'ONM exploite ces deux dispositifs pour numériser, mémoriser et afficher des formes d'ondes analogiques. Un tel oscilloscope présente de nombreux avantages si on le compare à un oscilloscope classique qui conserve l'image de la forme d'onde dans un écran grillagé interposé entre le canon électronique du tube et l'écran recouvert de poudres phosphores. En voici d'ailleurs une brève énumération:

- L'ONM mémorise les formes d'ondes de façon permanente parce que ce sont des données nunériques qui sont conservées dans des circuits numériques (par ex. des bascules). Dans l'oscilloscope courant, l'image s'estompe graduellement à mesure que l'écran grillagé se neutralise progressivement.

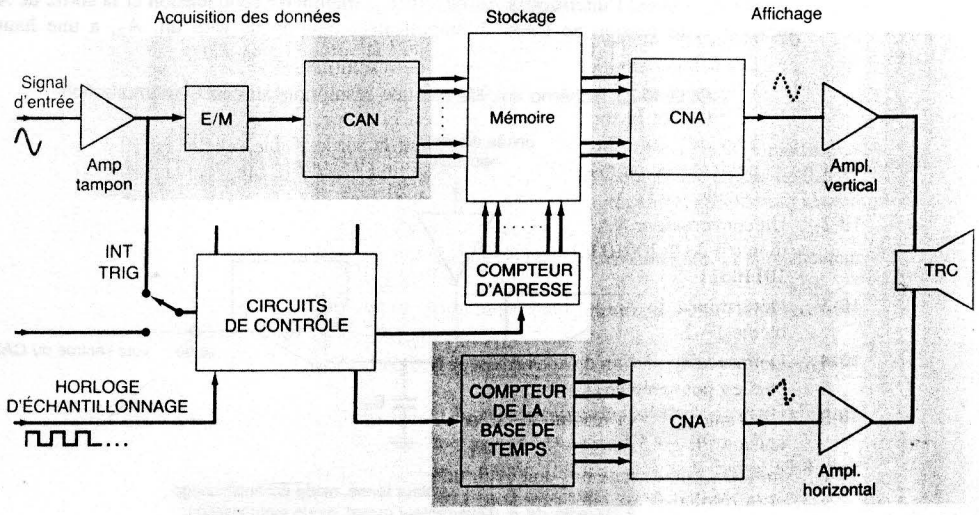
- Dans un ONM, la forme d'onde en mémoire peut être placée comme il nous plaît sur l'écran du tube cathodique et l'on peut régler les échelles verticale et horizontale en conséquence. Cette possibilité est inexistante dans un oscilloscope classique.
- Dans un ONM, il est possible de mémoriser et d'afficher des parties de la forme d'onde qui existait avant le point de déclenchement. De nouveau, une telle possibilité est irréalisable avec l'autre type d'oscilloscope.

La figure 10-25 nous fait voir le schéma fonctionnel d'un ONM. L'ensemble du fonctionnement est contrôlé et synchronisé par les circuits du bloc CONTRÔLE, en réalité un microprocesseur qui exécute un programme stocké dans une mémoire morte. La partie acquisition des données contient un circuit échantillonnage/mémorisation (E/M) et un CAN qui échantillonne et numérise le signal d'entrée à un rythme déterminé par l'HORLOGE D'ÉCHANTILLONNAGE; ces circuits envoient les données numérisées à la mémoire en vue de leur stockage. Le bloc de CONTRÔLE s'assure que les points successifs du signal sont mémorisés dans des emplacements mémoires successifs en mettant sans cesse à jour le COMPTEUR D'ADRESSE.

Quand la mémoire est garnie au complet, le point de donnée suivant provenant du CAN est mémorisé dans le premier emplacement du bloc mémoire, écrasant l'ancienne donnée, et ainsi de suite pour les autres points. Le processus de l'acquisition et du stockage des données se poursuit tant que le bloc de CONTRÔLE ne reçoit pas un signal de déclenchement INTERNE (de l'onde d'entrée) ou EXTERNE. Quand le déclenchement survient, le système cesse l'acquisition des données et se met dans le mode d'affichage, durant lequel une partie ou la totalité du contenu de la mémoire est affichée sans interruption sur l'écran cathodique.

Pour l'affichage, on retrouve dans l'appareil deux CNA qui fournissent les tensions de déviation verticale et horizontale. Les données conservées en mémoire causent la déviation verticale du faisceau d'électrons, tandis que le COMPTEUR DE LA BASE DE TEMPS

FIGURE 10-25 Schéma fonctionnel d'un oscillateur numérique à mémoire.



assure sa déviation horizontale, sous la forme d'un signal en escalier. Le bloc de CONTRÔLE synchronise l'affichage de l'onde en incrémentant simultanément le REGISTRE D'ADRESSE et le COMPTEUR DE LA BASE DE TEMPS, de façon qu'à chaque déplacement horizontal du faisceau corresponde au nouveau point en mémoire converti par le CNA. Les compteurs sont sans cesse réinitialisés et le traçage des données est répété sur l'écran. L'affichage consiste en une nuée de petits points représentant chaque donnée échantillonnée; comme cette nuée contient généralement un grand nombre de points sur l'écran (au-delà de 1000), ils semblent se toucher, et la forme d'onde prend l'aspect d'une ligne continue.

L'opérateur met un terme à l'affichage au moyen d'un bouton du panneau avant qui donne à l'ONM l'ordre de commencer l'acquisition d'un nouveau cycle de données.

QUESTIONS DE RÉVISION

1. Indiquez quelques avantages de l'ONM par rapport à l'oscilloscope classique.
2. Décrivez le rôle des CNA et CAN que l'on trouve dans un ONM.

PROBLÈMES

SECTIONS 10-1 et 10-2

10-1 CONTRÔLE

a) Quelle formule donne la relation entre la sortie et les entrées d'un CNA?
b) Définissez pas de progression d'un CNA.
c) Définissez résolution d'un CNA.
d) Définissez pleine échelle.
e) Donnez la définition de pourcentage de résolution.
f) *Vrai ou faux:* Un CNA de 10 bits a une résolution plus fine qu'un CNA de 12 bits dont la sortie pleine échelle est la même.
g) *Vrai ou faux:* Un CNA de 10 bits dont la sortie pleine échelle est 10 V a un pourcentage de résolution plus petit qu'un CNA de 12 bits dont la sortie pleine échelle est 12 V.

10-2 Un convertisseur NA de 8 bits produit une tension de sortie de 2,0 V pour le code d'entrée de 01100100. Dites quelle est la valeur de V_{SORTIE} pour le code d'entrée 10110011.

10-3 Déterminez le poids de chaque bit d'entrée du convertisseur NA du problème 10-2.

10-4 Donnez la résolution du convertisseur NA du problème 10-2. Exprimez-la en volts puis en pourcentage.

10-5 Dites quelle est la résolution en volts d'un convertisseur NA de 10 bits dont la tension PE est 5 V.

10-6 Indiquez combien de bits doit avoir un convertisseur dont l'intensité PE est 10 mA et la résolution est inférieure à 40 μA.

10-7 Dites quelle est la résolution en pourcentage PE du convertisseur NA de la figure 10-26. Indiquez le pas de progression si la tension sur l'échelon supérieure est 2 V.

10-8 Donnez la raison des pointes négatives de la forme d'onde V_{SORTIE} de la figure 10-26. (*Suggestion:* notez que le compteur est à propagation et que les pointes se produisent à chaque deux échelons.)

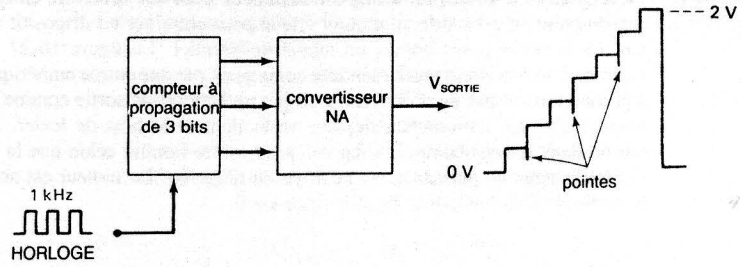


FIGURE 10-26

10-9 Soit un CNA parfaitement exact de 12 bits, à quelle vitesse fera-t-il tourner le moteur de la figure 10-4 que l'on veut régler à 250 tr/min?

10-10 Un convertisseur NA de 12 bits (3 chiffres) dont le code d'entrée est en DCB a une tension pleine échelle de 9,99 V. Déterminez le pas de progression, la résolution en pourcentage PE et la valeur de V_{SORTIE} pour le code d'entrée 0110 1001 0101.

10-11 Dites pour lequel des circuits suivants le pourcentage PE, exprimant la résolution, est le plus élevé: un convertisseur NA dont le code d'entrée est un nombre binaire de 8 bits ou un convertisseur dont l'entrée est une représentation DCB de 8 bits, la pleine échelle pour chacun est 990 mV.

10-12 Déterminez le poids de chaque bit du CNA DCB du problème 10-10. D'après vous, quelle sera la sortie si l'entrée est 100000100011?

SECTION 10-3

10-13 Le pas de progression du convertisseur NA de la figure 10-6 peut être modifié en changeant la valeur de R_F. Déterminez la valeur de R_F qui donne un pas de 0,5 V. Dites si cette nouvelle valeur de R_F change le pourcentage PE exprimant la résolution.

10-14 Supposez que la sortie du CNA de la figure 10-8 a) soit connectée à l'amplificateur opérationnel de la figure 10-8 b).

a) Si $V_{REF} = 5$ V, $R = 20$ kΩ et $R_F = 10$ kΩ, déterminez le pas de progression et la tension pleine échelle de V_{SORTIE}.
b) Changez la valeur de R_F pour que la tension pleine échelle en V_{SORTIE} soit -2 V.
c) Utilisez cette nouvelle valeur de R_F et déterminez le facteur de proportionnalité, K, de l'équation $V_{SORTIE} = K(V_{REF} \times B)$.

10-15 Quelle est la supériorité du CNA de la figure 10-9 sur celui de la figure 10-8, tout particulièrement quand le nombre de bits d'entrée est élevé.

SECTIONS 10-4 et 10-5

10-16 Un convertisseur NA de 8 bits a une erreur pleine échelle de 0,2 %. Si le CNA a une intensité pleine échelle de 10 mA, dites quelle est l'erreur maximale pour n'importe quelle entrée numérique. Si on lit à la sortie du CNA 50 μA pour une entrée numérique de 00000001, dites si on est dans la plage de précision spécifiée. (erreur de décalage nulle)

10-17 La régulation d'un dispositif de positionnement peut être effectuée au moyen d'un *servomoteur*, c'est-à-dire un moteur étudié pour entraîner un dispositif mécanique tant qu'il existe à ses bornes un signal différentiel. La figure 10-27 montre un système d'asservissement élémentaire commandé par une entrée numérique fournie, soit directement par un ordinateur, soit par un support de sortie comme une bande magnétique. Le servomoteur déplace verticalement le bras de levier. Le moteur tourne dans le sens horaire ou dans le sens contre-horaire selon que la tension de l'amplificateur de puissance est positive ou négative. Le moteur est arrêté quand la sortie de l'amplificateur de puissance est 0.

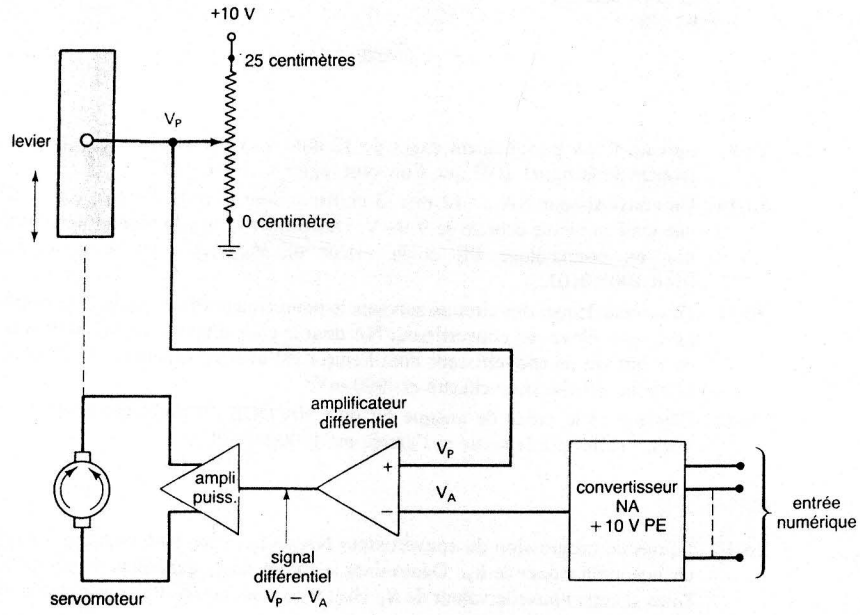


FIGURE 10-27

La position mécanique du levier est convertie en une tension continue par le potentiomètre illustré. Quand le levier est à son point de référence nul, $V_P = 0$ V. La valeur de V_P augmente au rythme de 1 V/2,5 cm jusqu'à ce que le levier atteigne son point le plus haut (25 cm) et que $V_P = 10$ V. La position recherchée pour le levier est fournie sous forme numérique par l'ordinateur à un convertisseur NA, qui donne V_A. La *différence* entre V_P et V_A (appelée *signal différentiel*) est produite par l'amplificateur *différentiel* et amplifiée par l'amplificateur de puissance pour faire tourner le moteur dans le sens qui diminue le signal différentiel jusqu'à 0, c'est-à-dire qui déplace le levier jusqu'à ce que $V_P = V_A$.

a) Si on veut déterminer une certaine position du levier avec une résolution de 0,25 cm, dites combien de bits doit avoir l'entrée numérique.
b) En fonctionnement réel, le bras de levier peut osciller légèrement autour de la position recherchée, spécialement si on utilise un potentiomètre *bobiné*. Expliquez.

10-18 CONTRÔLE

a) Définissez un réseau de résistances pondérées binairement.
b) Définissez un réseau $R/2R$.
c) Définissez temps d'établissement d'un CNA
d) Définissez erreur pleine échelle.
e) Définissez erreur de décalage.

10-19 Un certain CNA de 6 bits a une sortie pleine échelle de 1,260 V. Son exactitude est spécifiée par une précision de $\pm 0{,}1$ % P.E. et une erreur de décalage de ± 1 mV. Supposons que l'erreur de décalage n'a pas été compensée à zéro. Soit les mesures suivantes réalisées sur ce CNA; déterminez laquelle ne correspond pas à une des spécifications de l'appareil. (*Suggestion:* L'erreur de décalage est ajoutée à l'erreur introduite par les imprécisions propres des composants.)

Code d'entrée	Sortie (mV)
000010	41,5
000111	140,2
001100	242,5
111111	1,258 V

SECTION 10-6

10-20 Un certain CNA possède les données techniques suivantes: résolution 8 bits, pleine échelle = 2,55 V, erreur $\leq$ 2 mV; exactitude = $\pm 0{,}1$ % P.E. Un essai en régime statique de cet élément produit les résultats suivants. Quelle est la cause probable de ce mauvais fonctionnement?

Code d'entrée	Sortie
00000000	8 mV
00000001	18,2 mV
00000010	28,5 mV
00000100	48,3 mV
00001111	158,3 mV
10000000	1,289 V

10-21 Même question qu'au problème 10-21, mais cette fois-ci pour les données suivantes:

Code d'entrée	Sortie
00000000	20,5 mV
00000001	30,5 mV
00000010	20,5 mV
00000100	60,6 mV
00001111	150,6 mV
10000000	1,300 V

10-22 Un technicien raccorde un compteur au CNA de la figure 10-3 en vue de réaliser un essai en escalier avec un horloge de 1 kHz. Le résultat de son expérience est reproduit à la figure 10-28. Quelle est la cause probable de ce signal en escalier erroné?

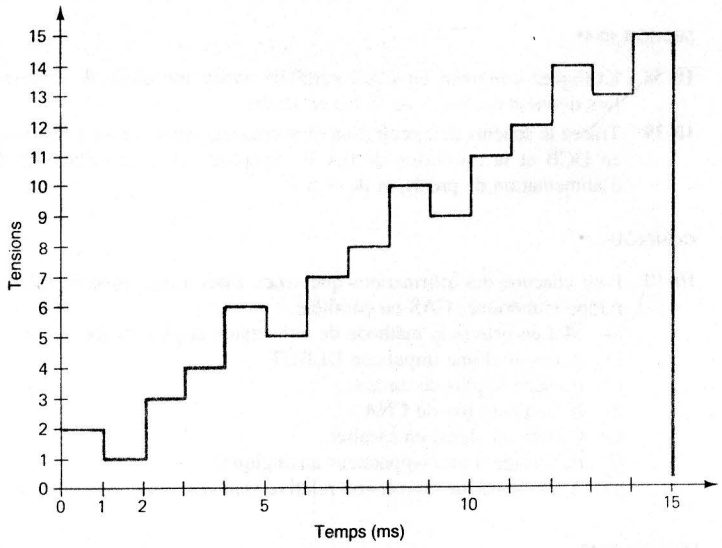


FIGURE 10-28

SECTIONS 10-7 et 10-8

10-23 CONTRÔLE

Voici une description du fonctionnement d'un CNA rampe numérique. Inscrivez le ou les mots manquants.

Une impulsion DÉBUT est appliquée afin de _____ le compteur et d'empêcher _____ de traverser la porte ET et de se rendre dans _____ . À ce moment-ci, la sortie du CNA, V_{AX} est _____ et le signal $\overline{FDC}$ est _____ .

Au moment où DÉBUT revient _____ , la porte ET _____ , le compteur peut désormais _____ . Le signal V_{AX} progresse d'un _____ à la fois jusqu'à ce _____ V_A. Quand cela se produit, _____ passe au niveau BAS pour _____ les autres impulsions provenant de _____ . Ceci indique la fin de la conversion, l'équivalent numérique de V_{AX} est présent dans _____ .

10-24 Un convertisseur AN rampe numérique de 8 bits ayant une résolution de 40 mV a une fréquence d'horloge de 2,5 mHz et un comparateur dont V_T = 1 mV. Calculez:

a) La sortie numérique pour V_A = 6,000 V.
b) La sortie numérique pour 6,035 V.
c) Les durées de conversion maximale et moyenne de ce convertisseur.

10-25 Expliquez pourquoi les sorties numériques pour les parties a) et b) du problème 10-24 sont égales.

10-26 Dites ce qui se passe dans le convertisseur AN du problème 10-24 si une tension analogique de V_A = 10,853 V est appliquée à son entrée. Tracez la forme d'onde que l'on observe en sortie. Ajoutez au convertisseur AN les portes logiques nécessaires pour qu'une indication « hors échelle » soit produite quand V_A est trop grand.

10-27 Soit les caractéristiques suivantes d'un convertisseur AN: résolution, 12 bits; erreur pleine échelle, 0,03 % PE; tension pleine échelle, +5 V.

a) Indiquez l'erreur de quantification en volts.
b) Donnez l'erreur totale possible en volts.

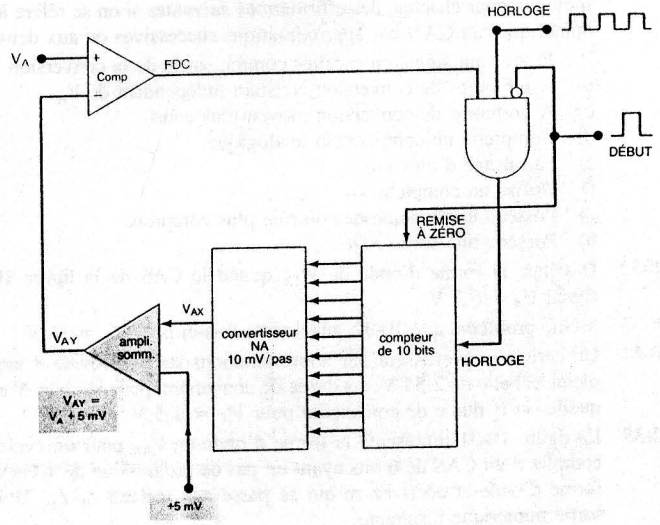
10-28 L'erreur de quantification d'un convertisseur AN, comme celui de la figure 10-12, est toujours positive puisque V_{AX} doit dépasser V_A pour que la sortie du COMPARATEUR change d'état. Ceci veut dire que la valeur de V_{AX} peut dépasser V_A d'au plus un poids faible. Il est possible de modifier cette erreur de quantification de façon que V_{AX} soit égale à V_A à $\pm 1/2$ poids faible près. Pour ce faire, on ajoute une tension fixe égale à 1/2 poids faible (1/2 pas de progression) à la valeur V_{AX}. La figure 10-29 montre cela symboliquement pour un convertisseur ayant une résolution de 10 mV/pas. Une tension fixe de +5 mV est ajoutée à la sortie NA de l'amplificateur de sommation et le résultat, V_{AY}, est délivré au comparateur dont V_T = 1 mV. Déterminez pour ce convertisseur modifié la sortie numérique quand:

a) V_A = 5,011 V
b) V_A = 5,028 V

Déterminez l'erreur de quantification dans chaque cas en comparant V_{AX} et V_A. Notez que l'erreur est positive dans un cas et négative dans l'autre.

10-29 Pour le CAN de la figure 10-29, trouvez la plage des entrées analogiques qui donne en sortie la valeur numérique 0100011100.

FIGURE 10-29



SECTION 10-9

10-30 Supposons que le signal analogique de la figure 10-30 soit numérisé au moyen d'une conversion continuelle par un convertisseur rampe numérique dont l'escalier prend de la hauteur au rythme de 1 V à toutes les 25 μs. Tracez un croquis du signal reconstitué au moyen des données obtenues par le processus de numérisation. Comparez ce croquis au signal original, indiquez ce qui pourrait être fait pour rendre cette représentation plus précise.

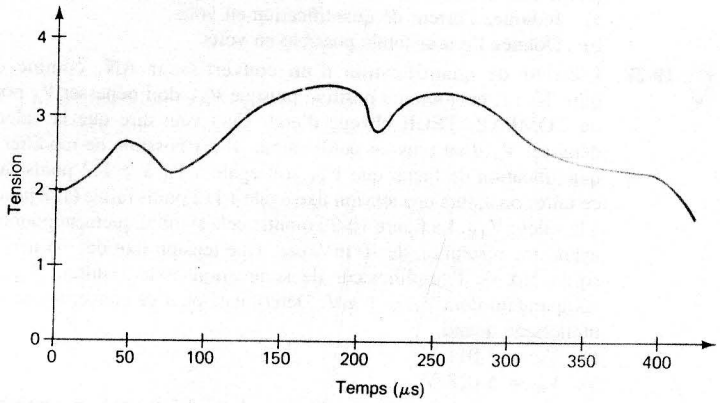


FIGURE 10-30

SECTION 10-10

10-31 CONTRÔLE

Indiquez pour chacune des affirmations suivantes si on se réfère à un CAN rampe numérique, un CAN par approximations successives ou aux deux.

a) Produit un signal en escalier comme sortie de la conversion NA.
b) A un temps de conversion constant indépendant de V_A.
c) A un temps de conversion moyen plus court.
d) Comprend un comparateur analogique.
e) Fait usage d'un CNA.
f) Utilise un compteur.
g) Possède une logique de contrôle plus complexe.
h) Possède une sortie $\overline{FDC}$.

10-32 Dessinez la forme d'onde de V_{AX} quand le CAS de la figure 10-7 convertit le signal $V_A = 6{,}7$ V.

10-33 Même problème que 10-32, mais cette fois-ci pour $V_A = 16$ V.

10-34 Un certain convertisseur par approximations successives de 8 bits a une tension pleine échelle de 2,55 V. La durée de conversion pour $V_A = 1$ V est 80 μs. Dites quelle est la durée de conversion pour $V_A = 1{,}5$ V.

10-35 La figure 10-31 représente la forme d'onde en V_{AX} pour un cycle de conversion complet d'un CAS de 6 bits ayant un pas de progression de 40 mV. Étudiez cette forme d'onde et décrivez ce qui se passe aux instants t_0–t_5. Trouvez ensuite la sortie numérique résultante.

10-36 Reportez-vous à la figure 10-19. Quelle est la valeur approximative de l'entrée analogique si le bus de donnée du micro-ordinateur contient 10010111 à l'instant où $\overline{RD}$ est porté momentanément vers le BAS?

10-37 Connectez une référence de tension de 2,0 V sur la borne $V_{réf}/2$ et refaites le problème 10-36.

SECTION 10-11

10-38 Expliquez comment un CAN parallèle ayant une durée de conversion de 1 μs fonctionne dans le cas de la figure 10-30.

10-39 Tracez le schéma de circuit d'un convertisseur parallèle de 4 bits dont la sortie est en DCB et la résolution de 0,1 V. Supposez que vous disposez d'une tension d'alimentation de précision de +5 V.

CONTRÔLE

10-40 Pour chacune des affirmations que voici, dites à quel type de CAN on se réfère: rampe numérique, CAS ou parallèle.

a) Met en oeuvre la méthode de conversion la plus rapide.
b) A besoin d'une impulsion DÉBUT.
c) Possède le plus de circuits.
d) Ne contient pas de CNA.
e) Génère un signal en escalier.
f) Fait usage d'un comparateur analogique.
g) A une durée de conversion relativement constante quel que soit V_A.

SECTION 10-12

CONTRÔLE

10-41 Pour chacune des affirmations suivantes, dites à quel CAN on se réfère:

a) Possède un compteur qui n'est jamais remis à zéro.
b) A recours à un très grand nombre de comparateurs.
c) Utilise un *VCO*.
d) Est principalement employé dans les milieux industriels parasités.
e) Exploite les caractéristiques d'un condensateur.
f) Est relativement peu affecté par la température.

SECTION 10-13

10-42 On applique une tension $V_A = 3{,}853$ V au voltmètre numérique de la figure 10-21. Présumez que les compteurs sont initialement à 0 et tracez les formes d'ondes en V_{AX}, COMP, Q_1, Q_2 et à la sortie de la porte ET pour deux cycles complets. Utilisez un signal d'horloge de 100 kHz et supposez que les monostables ont $t_p = 1$ μs. V_A arrive à 3,86 V (386 pas) avant que COMP devienne BAS.

10-43 Reportez-vous au circuit du voltmètre numérique de la figure 10-21. Supposez que ce circuit fonctionne correctement pour les valeurs V_A ne dépassant pas 7,99 V. Toutefois, quand V_A est porté au-delà de 8 V, la valeur affichée ne change pas, la forme d'onde en escalier de V_{AX} atteint seulement 7,99 V environ avant de tomber à 0, et COMP demeure au niveau HAUT. Dites quelles sont les causes probables de ce mauvais fonctionnement.

10-44 Ajoutez au voltmètre numérique de la figure 10-21 la fonction logique nécessaire pour disposer d'un voyant DEL hors calibre qui s'allume dès que la tension analogique d'entrée dépasse 9,99 V. Ce voyant doit s'éteindre au début d'une nouvelle conversion.

SECTIONS 10-14 et 10-15

10-45 Reportez-vous au circuit échantillonnage et mémorisation de la figure 10-23. Indiquez quelle défaillance du circuit produit en V_{SORTIE} une onde exactement semblable à V_A. Dites quelle défaillance garde toujours V_{SORTIE} à 0.

10-46 Utilisez dans la figure 10-24 les interrupteurs bilatéraux S_1, S_2 et S_3 du CMOS 4016 et concevez la logique de contrôle nécessaire pour que chaque entrée analogique soit convertie séquentiellement en son équivalent numérique. Le convertisseur AN est un convertisseur par approximations successives de 10 bits fonctionnant avec un signal d'horloge de 50 kHz et qui a besoin d'une impulsion d'une durée de 10 μs pour commencer la conversion. Les sorties numériques doivent demeurer stables après l'achèvement de la conversion avant de passer à l'entrée analogique suivante. Choisissez la fréquence de l'horloge de multiplexage appropriée.

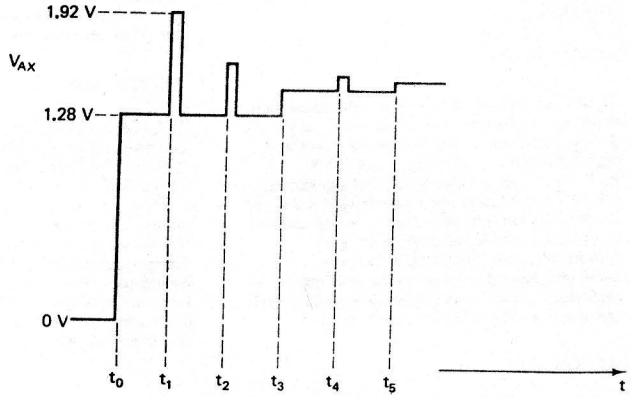


FIGURE 10-31

APPLICATIONS AU MICROPROCESSEUR

10-47 La figure 10-19 nous montre comment un ADC8004 est relié à un micro-ordinateur. On y voit trois signaux de commande, $\overline{CS}$, $\overline{RD}$, et $\overline{WR}$ qui proviennent du micro-ordinateur et aboutissent au CAN. Ces signaux ont pour rôle de lancer une nouvelle conversion AN et de transférer la donnée de sortie du CAN sur le bus de donnée du micro-ordinateur.

La figure 10-32 a) illustre une façon de produire ces signaux dans les circuits logiques de décodage d'adresse et de contrôle du micro-ordinateur. Le microprocesseur (MP) de cet ordinateur possède une seule sortie $R/\overline{W}$ qui signale au MP qu'il s'agit d'une opération de lecture ou d'une opération d'écriture. Quand il s'agit d'une lecture $R/\overline{W} = 1$, pour le cas d'une écriture $R/\overline{W} = 0$.

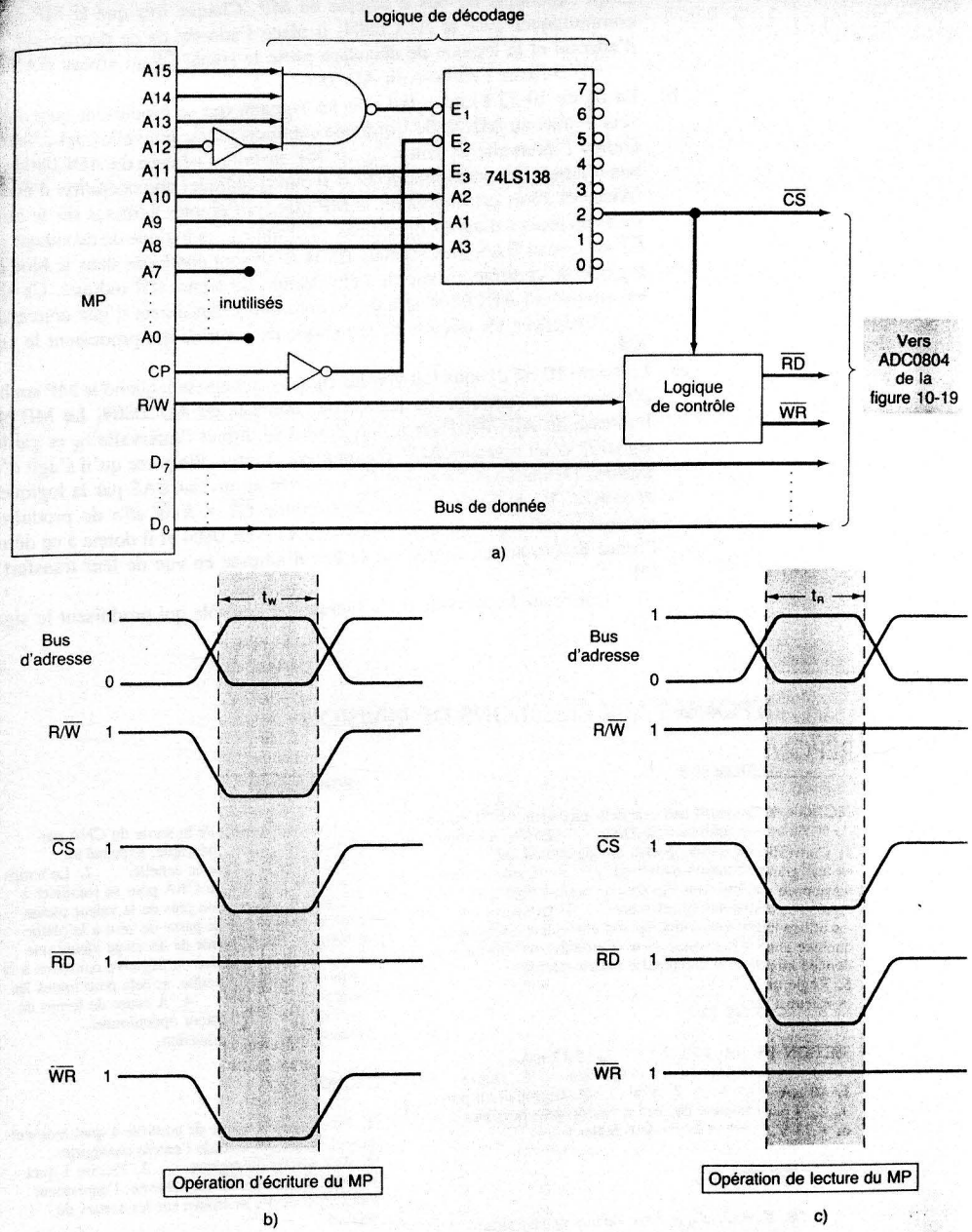


FIGURE 10-32 a) Le MP est mis en liaison avec le ADC0804 de la figure 10-19; b) signaux nécessaires pour une opération d'écriture par le MP; c) signaux pour une opération de lecture par le MP.

a) Le signal $\overline{CS}$ qui valide le ADC0804 est produit par les huit conducteurs de rangs supérieurs du bus d'adresse du MP. Chaque fois que le MP souhaite communiquer avec le ADC0804, il place l'adresse de ce dernier sur le bus d'adresse et la logique de décodage porte le signal $\overline{CS}$ au niveau BAS.

Établissez l'adresse du ADC0804.

b) La figure 10-32 b) nous fait voir les signaux qui se produisent quand le MP veut donner au ADC0804 l'ordre de commencer une nouvelle conversion AN. Durant l'intervalle de temps t_W, le MP installe l'adresse du ADC0804 sur le bus d'adresse et positionne $R/\overline{W} = 0$ afin d'indiquer une opération d'écriture. (Avant et après cet intervalle, le MP place différentes adresses sur le bus qui sont destinées à d'autres opérations.) Durant t_W, la logique de décodage garde $\overline{CS}$ au niveau BAS. Les signaux $\overline{CS}$ et $R/\overline{W}$ sont combinés dans le bloc de la logique de contrôle en vue de l'élaboration du signal $\overline{WR}$ indiqué. Ce signal est envoyé au ADC0804 afin de lui signifier le lancement d'une conversion.

Concevez les circuits de la logique de contrôle qui produisent le signal $\overline{WR}$.

c) La figure 10-32 c) nous fait voir les signaux qui agissent quand le MP souhaite effectuer une opération de lecture des données du ADC0804. Le MP place l'adresse du ADC0804 sur le bus d'adresse durant l'intervalle t_R et garde le signal $R/\overline{W}$ au niveau HAUT durant tout cet intervalle parce qu'il s'agit d'une lecture. De nouveau, le signal $\overline{CS}$ est porté au niveau BAS par la logique de décodage. La logique de contrôle combine $\overline{CS}$ et $R/\overline{W}$ afin de produire le signal $\overline{RD}$ indiqué. Ce signal est envoyé au ADC0804 et il donne à ce dernier l'ordre de placer ses sorties sur le bus d'adresse en vue de leur transfert au MP.

Concevez les circuits de la logique de contrôle qui produisent le signal $\overline{RD}$.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 10-1

1. Convertit une grandeur physique non électrique en une grandeur électrique. **2.** Convertit une tension ou un courant analogique en une représentation numérique. **3.** Les mémorise; les utilise dans des calculs ou leur applique quelque autre traitement. **4.** Convertit une donnée numérique en son équivalent analogique. **5.** Règle une grandeur physique conformément à la valeur d'un signal d'entrée électrique.

SECTION 10-2

1. 40 μA; 10,2 mA. **2.** 5,12 mA. **3.** 0,39 pour cent. **4.** 4096. **5.** Douze. **6.** 8,73 V. **7.** Vrai. **8.** Il produit un plus grand nombre de sorties analogiques possibles entre zéro et la pleine échelle.

SECTION 10-3

1. N'emploie que deux valeurs de résistance différentes. **2.** 640 kΩ. **3.** 0,5 V.

SECTION 10-4

1. L'écart maximal de la sortie du CNA par rapport à sa valeur théorique, exprimé en pourcentage de la pleine échelle. **2.** Le temps que prend la sortie du CNA pour se stabiliser à 1/2 pas de progression près de la valeur pleine échelle quand l'entrée passe de zéro à la pleine échelle. **3.** L'erreur de décalage ajoute une petite déviation positive ou négative constante à la sortie analogique attendue, et cela pour toutes les entrées numériques. **4.** À cause du temps de réponse de l'amplificateur opérationnel convertisseur courant-tension.

SECTION 10-7

1. Précise à la logique de contrôle à quel moment la sortie du CAN excède l'entrée analogique. **2.** Aux sorties du registre. **3.** Précise à quel moment la conversion est achevée; l'équivalent numérique de V_A se trouve sur les sorties du registre.

SECTION 10-8

1. L'entrée numérique d'un CNA est incrémentée tant que la sortie en escalier de l'appareil ne dépasse pas la valeur de l'entrée analogique. **2.** L'erreur propre imputable au fait que V_{AX} ne progresse pas continûment, mais procède par pas égaux à la résolution du CNA. Le V_{AX} final peut différer de V_A par au plus une hauteur d'échelon (un pas de progression). **3.** Si V_A augmente, il faut plus d'échelons à V_{AX} pour l'atteindre et le dépasser. **4.** Vrai. **5.** Circuit peu complexe; durée de conversion relativement longue et dépendante de V_A. **6.** $00100000111_2 = 135_{10}$ pour les deux cas.

SECTION 10-9

1. Processus de conversion de différents points d'un signal analogique en des données numériques et mémorisation de ces dernières. **2.** L'ordinateur génère un signal DÉBUT pour initier la conversion AN. Quand $\overline{FDC}$ passe à 0, cette transistion est l'ordre lancé à l'ordinateur que la conversion est achevée. Ce dernier charge la sortie numérique du CAN en mémoire. Ce processus est répété pour le point suivant du signal analogique.

SECTION 10-10

1. Le CAS possède une durée de conversion plus courte qui est indépendante de la valeur de V_A. **2.** Il possède des circuits logiques de contrôle plus complexes. **3.** Faux. **4. a)** 8; **b)** 0-5 V; **c)** $\overline{CS}$ fixe le rôle des signaux $\overline{RD}$ et $\overline{WR}$, $\overline{WR}$ sert à lancer une nouvelle conversion, $\overline{RD}$ à valider les tampons de sortie. **d)** Quand elle est BASSE, c'est le signal de la fin de la conversion; **e)** Pour séparer la masse numérique généralement parasitée de la masse analogique et ainsi s'assurer que le signal d'entrée analogique n'est pas déformé par le bruit.

SECTION 10-11

1. Vrai. **2.** 4095 comparateurs et 4096 résistances. **3.** Sa principale supériorité est sa vitesse de conversion; son inconvénient, le nombre de composants de circuit nécessaire pour atteindre à une précision acceptable.

SECTION 10-12

1. Il accélère la conversion au moyen d'un compteur réversible (compteur/décompteur) qui permet à V_{AX} de poursuivre V_A sans devoir revenir à zéro au début de chaque nouvelle conversion. **2.** Un VCO. **3.** Avantages: peu coûteux, insensibilité thermique; inconvénient: longue durée de conversion. **4.** CAN parallèle, CAN tension-fréquence, CNA à double rampe.

SECTION 10-13

1. Ils mémorisent en vue d'un affichage les valeurs numériques présentes à la sortie des compteurs à la fin de la conversion. **2.** Le FM du monostable 1 transfère le résultat de la conversion aux registres; l'impulsion émanant du monostable 2 remet à zéro les compteurs avant le nouveau cycle de conversion.

SECTION 10-14

1. Il prélève un échantillon de la tension analogique d'entrée et le mémorise dans un condensateur. **2.** Faux; il s'agit de tampons de gain unité ayant une haute impédance d'entrée et une petite impédance de sortie.

SECTION 10-16

1. La forme d'onde affichée peut être déplacée et manipulée; il est possible d'afficher des parties de la forme d'onde qui était affichée avant le déclenchement; la durée de conservation en mémoire est infinie. **2.** Le CAN numérise les points de la forme d'onde d'entrée en vue de leur mémorisation; le CNA vertical convertit les données ponctuelles mémorisées en valeurs analogiques pour obtenir la déviation verticale du faisceau d'électrons; le CNA horizontal produit une tension de balayage en escalier qui assure la déviation horizontale du faisceau d'électrons.

11

DISPOSITIFS MÉMOIRES

Contenu

11-1 Terminologie des mémoires
11-2 Fonctionnement d'une mémoire: généralités
11-3 Connexions UCT-mémoire
11-4 Mémoires MORTES
11-5 Architecture des mémoires MORTES
11-6 Synchronisation des mémoires MORTES
11-7 Types de mémoires MORTES
11-8 Applications des mémoires MORTES
11-9 Éléments logiques programmables
11-10 Mémoires VIVES à semiconducteurs
11-11 Architecture des mémoires VIVES
11-12 Mémoires VIVES statiques (SRAM)
11-13 Mémoires VIVES dynamiques (DRAM)
11-14 Structure et fonctionnement des mémoires VIVES dynamiques
11-15 Cycles de lecture/écriture des mémoires VIVES dynamiques
11-16 Rafraîchissement des mémoires VIVES dynamiques
11-17 Extension de la longueur du mot et de la capacité
11-18 Mémoires VIVES rémanentes
11-19 Mémoires séquentielles
11-20 Mémoires magnétiques
11-21 Dépannage des systèmes de mémoires VIVES
11-22 Essai des mémoires MORTES

Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Comprendre et employer à bon escient la terminologie des systèmes de mémoires.
- Décrire la différence entre une mémoire à lecture/écriture et une mémoire à lecture seulement (dite inaltérable).
- Décrire la différence entre une mémoire à accès séquentiel et une mémoire à accès sélectif.
- Établir la capacité d'un dispositif mémoire d'après ses entrées et ses sorties.
- Donner les grandes lignes de l'enchaînement des étapes quand l'UCT lit ou écrit dans une mémoire.
- Faire la distinction entre les différents types de mémoires MORTES et donner des exemples d'utilisation.
- Comprendre et décrire la structure et l'organisation des mémoires VIVES statiques et dynamiques.
- Combiner des mémoires intégrées afin d'étendre la capacité de la mémoire.
- Décrire l'architecture des éléments logiques programmables de base et trouver la configuration des fusibles correspondant à un certain ensemble de fonctions.
- Décrire les caractéristiques élémentaires des différents types de mémoires magnétiques.
- Utiliser les résultats de mesure sur des mémoires VIVES et MORTES en vue de déceler les défectuosités possibles.

Introduction

Les systèmes numériques disposent d'un avantage important par rapport aux systèmes analogiques, soit leur aptitude à mémoriser facilement de grandes quantités de données et d'informations numériques aussi bien à long terme qu'à court terme. C'est la capacité de mémorisation qui explique la polyvalence des systèmes numériques et leur adaptabilité à de nombreuses situations. Par exemple, la mémoire principale interne d'un ordinateur conserve des instructions qui indiquent à l'ordinateur ce qu'il faut faire dans *toutes* les circonstances possibles pour que le travail fait sous la gouverne de l'ordinateur exige le moins d'interventions humaines possibles.

Le chapitre que vous abordez présentement fait l'examen des types de dispositifs mémoires les plus courants. Vous êtes déjà très familiarisés avec un dispositif mémoire

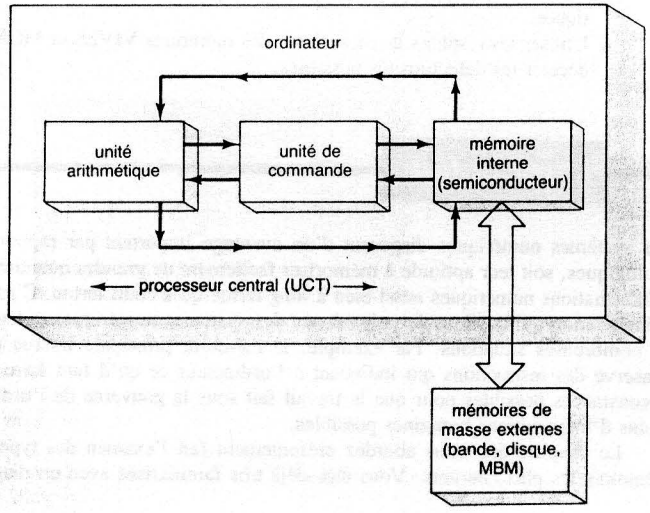
électronique, la bascule. Vous avez vu des groupes de bascules, appelés registres, qui mémorisent de l'information et transfèrent cette dernière dans d'autres emplacements. Les registres à bascules sont des éléments de mémoire rapides que les opérations internes d'un ordinateur utilisent intensivement, car dans ces opérations l'information numérique est sans arrêt déplacée d'un endroit à un autre. Les bonds prodigieux accomplis par la technologie LSI et VLSI rendent maintenant possible l'intégration sur une simple puce d'un très grand nombre de bascules, agencées selon divers formats de matrices mémoires. Les mémoires à semiconducteurs bipolaires et MOS sont les dispositifs les plus rapides sur le marché, et leur coût n'a cessé de décroître à mesure que la technologie LSI franchissait de nouveaux sommets.

Les données numériques peuvent également être stockées sous la forme de charges dans des condensateurs; d'ailleurs un type de mémoire à semiconducteurs très important exploite ce principe pour obtenir le stockage de beaucoup d'informations dans peu d'espace tout en dissipant une très faible puissance.

Les mémoires à semiconducteurs se retrouvent surtout dans la *mémoire interne* de l'ordinateur (figure 11-1) là où la vitesse d'opération est importante. La mémoire interne d'un ordinateur est constamment en communication avec l'unité centrale de traitement (UCT) lorsque les instructions d'un programme sont exécutées. Le programme, de même que les données utilisées par ce dernier, sont généralement stockés dans la mémoire interne.

Les mémoires à semiconducteurs sont taillées sur mesure pour la mémoire interne rapide; toutefois, il est impensable, à cause de leur coût par bit de mémoire, de les utiliser comme dispositifs de **mémoires de masse**. On appelle mémoire de masse une mémoire qui est extérieure à l'ordinateur principal (figure 11-1) et qui a la capacité de stocker des millions de bits *sans* consommer d'énergie électrique. La mémoire de masse est généralement plus lente que la mémoire interne, et c'est dans la mémoire de masse que l'on conserve l'information (programmes, données, etc.) dont n'a pas immédiatement besoin l'ordinateur. Cette information est transférée dans la mémoire interne lorsque l'ordinateur en a besoin.

FIGURE 11-1 Habituellement un système informatique a une mémoire interne très rapide et des mémoires de masse périphériques plus lentes.



Les mémoires de masse les plus répandues sont la bande magnétique et le disque magnétique dont le coût par bit est beaucoup plus faible que celui de la mémoire interne. Un nouveau venu dans la catégorie des mémoires de masse est la **mémoire à bulles magnétiques** (MBM), un dispositif à semiconducteurs qui exploite des principes magnétiques pour conserver des millions de bits par puce. Les MBM sont relativement lentes et ne peuvent servir de mémoires internes.

Avant d'étudier plus en détail les caractéristiques de ces dispositifs mémoires, définissons quelques-uns des termes importants que l'on rencontre fréquemment dans beaucoup de systèmes de mémorisation.

11-1 TERMINOLOGIE DES MÉMOIRES

L'étude des dispositifs mémoires et des systèmes de mémoires fait appel à une terminologie spéciale qui peut être déroutante pour un étudiant. Avant de se lancer dans une explication approfondie des mémoires, il serait bon de clarifier le sens de certains termes. La liste suivante n'est pas exhaustive; nous l'augmenterons au fur et à mesure que nous rencontrerons de nouvelles expressions.

Cellule mémoire: Circuit électrique ou dispositif qui stocke un seul bit (0 ou 1). Comme exemple de cellule mémoire, mentionnons la bascule, un condensateur, un point aimantable d'une bande ou d'un disque magnétique.

Mot mémoire: Groupe de bits (cellules) dans la mémoire qui correspond à des instructions ou à des données d'un certain type. Par exemple, un registre formé de huit bascules peut être considéré comme une case mémoire qui conserve un mot de 8 bits. La longueur des mots dans les ordinateurs d'aujourd'hui s'échelonne habituellement entre 4 et 64 bits selon la taille de l'ordinateur.

Octet: Expression spéciale désignant un mot de 8 bits. Un octet comme son nom l'indique a toujours 8 bits, et c'est la longueur de mot la plus courante dans les micro-ordinateurs.

Capacité: Nombre correspondant à la quantité de bits que l'on peut mémoriser dans un dispositif mémoire particulier ou dans un système mémoire complet. Par exemple, supposons une mémoire ayant 4096 mots de 20 bits, cela représente une capacité de mémorisation totale de 81 920 bits. Il est également possible d'exprimer cette capacité de mémorisation comme 4096 x 20. Dans cette dernière façon de parler, le premier nombre (4096) est le nombre de mots et le second nombre (20) est le nombre de bits par mot (longueur du mot). Le nombre de mots dans une mémoire est souvent un multiple de 1024. Il est courant en pratique d'appeler « 1 K » le chiffre 1024 lorsqu'on parle de la capacité de mémoire. Ainsi une mémoire qui a une capacité de 4K x 20 est en réalité une mémoire de 4096 x 20.

La mise au point de mémoires de plus en plus grandes nous a forcé à inventer une nouvelle désignation de la capacité: le mégaoctet ou « Mo », qui représente 2^{20} = 1 048 576 octets. Puisque un octet vaut 8 bits, une mémoire qui aurait une capacité de 2 Mo contient en réalité 2 097 152 emplacements de 8 bits.

EXEMPLE 11-1A

On indique qu'un boîtier mémoire à semiconducteurs est un 2K x 8. Dites combien on peut mémoriser de mots dans ce boîtier. Dites quelle est la longueur du mot. Indiquez le nombre total de bits que l'on peut placer dans ce boîtier.

Solution

$$2K = 2 \times 1024 = \mathbf{2048} \text{ mots}$$

Chaque mot a 8 bits (un octet). Par conséquent, le nombre total de bits est:

$$2048 \times 8 = \mathbf{16\ 384} \text{ bits}$$

EXEMPLE 11-1B

Laquelle de ces deux mémoires peut conserver le plus de bits: une mémoire de 5 Mo ou une mémoire de 1 mégamot de 16 bits de longueur?

Solution

$$5 \text{ Mo} = 5 \times 1\ 048\ 576 \times 8 = 41\ 943\ 040 \text{ bits}$$
$$1 \text{ mégamot} \times 16 = 1\ 048\ 576 \times 16 = 16\ 777\ 216 \text{ bits}$$

La réponse: la mémoire de 5 Mo.

Adresse: Nombre qui permet de repérer la localisation physique d'un mot en mémoire. Chaque mot stocké dans un dispositif ou dans un système mémoire possède une adresse exclusive. Les adresses sont toujours spécifiées sous forme de nombres, bien que l'on utilise pour plus de commodité leurs équivalents octal, hexadécimal ou décimal. La figure 11-2 illustre une petite mémoire formée de 8 mots. À chacun de ces 8 mots correspond une adresse particulière représentée par un nombre de 3 bits allant de 000 à 111. Chaque fois qu'on veut se reporter à l'emplacement en mémoire d'un mot particulier, on utilise son adresse pour le repérer.

Opération de lecture: Opération au cours de laquelle un mot binaire stocké dans un emplacement mémoire particulier (adresse) est retrouvé puis transféré dans un autre emplacement. Par exemple, si on a besoin du mot 4 de la mémoire de la figure 11-2 pour une raison quelconque, on effectue une opération de lecture de l'adresse 100. Une opération de lecture est souvent appelée une opération *d'extraction*, puisqu'un mot est extrait de la mémoire. Nous utiliserons ici indistinctement ces deux expressions.

Opération d'écriture: Opération au cours de laquelle un nouveau mot est installé dans un certain emplacement en mémoire. On donne aussi le nom d'opération de *stockage* à ce

FIGURE 11-2 À chaque localisation d'un mot est associée une adresse binaire exclusive.

adresses	
000	mot 0
001	mot 1
010	mot 2
011	mot 3
100	mot 4
101	mot 5
110	mot 6
111	mot 7

mouvement. Chaque fois qu'un nouveau mot est écrit dans un emplacement en mémoire, il remplace le mot s'y trouvant déjà.

Temps d'accès: Mesure de la vitesse de fonctionnement d'un dispositif mémoire. Il s'agit de la durée nécessaire pour effectuer une opération de lecture. Plus particulièrement, c'est le temps qui s'écoule entre le moment où la mémoire reçoit le signal de commande d'une lecture et le moment où la donnée se trouve sur la sortie de la mémoire. Le symbole désignant le temps d'accès est t_{ACC}.

Mémoire volatile: Type de mémoire qui ne conserve l'information stockée que s'il est sous une tension électrique. Si cette tension est enlevée, toute l'information stockée dans la mémoire est perdue. La majorité des mémoires à semiconducteurs sont volatiles, alors que la totalité des mémoires magnétiques sont rémanentes.

Mémoire à accès sélectif: Mémoire dans laquelle toutes les localisations physiques sont équivalentes pour ce qui est du temps de lecture ou d'écriture d'une donnée. Autrement dit, le temps d'accès est le même quelle que soit l'adresse de la mémoire. La plupart des mémoires à semiconducteurs sont des mémoires à accès sélectif.

Mémoire à accès séquentiel: Type de mémoire dans lequel le temps d'accès n'est pas constant mais varie en fonction de la localisation adressée. Un certain mot mémorisé est repéré par la lecture de la séquence de toutes les adresses jusqu'à ce que l'adresse recherchée soit atteinte. Ceci donne des temps d'accès beaucoup plus longs que ceux des mémoires à accès sélectif. Comme exemples de dispositifs à accès sélectif, mentionnons la bande mangétique, le disque magnétique et la mémoire à bulles magnétiques. Voici une illustration de la différence entre une mémoire à accès séquentiel et une mémoire à accès sélectif. Vous avez enregistré sur une cassette audio des chansons pendant 60 minutes. Si vous voulez jouer une chanson précise, vous devez faire avancer ou reculer la bande jusqu'à ce que vous trouviez votre chanson. C'est une solution relativement lente et le temps nécessaire pour retrouver la chanson dépend de sa place sur la bande. Voilà un type de mémorisation à accès séquentiel, parce qu'il faut examiner successivement toute l'information que l'on a lorsqu'on recherche quelque chose. La contrepartie à accès sélectif pourrait être le jukebox (ou le disque compact): vous voulez faire jouer une chanson, vous composez son code et le disque est retrouvé. Dans un jukebox, la durée de la recherche est la même pour toutes les chansons.

Mémoire VIVE: Mémoire dans laquelle il est aussi facile de lire que d'écrire une donnée.

Mémoire MORTE: Imposante catégorie de mémoires à semiconducteurs conçue pour les applications et dont le rapport du nombre d'opérations de lecture au nombre d'opérations d'écriture est très grand. Techniquement, les données dans une mémoire morte sont écrites (programmées) une seule fois, généralement en usine. Ensuite l'information qui s'y trouve peut seulement être lue. D'autres types de mémoires MORTES sont en réalité des mémoires presque inaltérables dans lesquelles on peut écrire des données plus d'une fois, mais en suivant une processus d'écriture beaucoup plus compliqué que l'opération de lecture courante; d'ailleurs, ce processus d'écriture est mis en oeuvre très rarement. Nous étudierons plus loin les divers types de mémoires MORTES. Toute les mémoires MORTES sont rémanentes et conservent leurs données même après coupure du courant.

Dispositifs mémoires statiques: Dispositifs à semiconducteurs qui conservent les données stockées tant qu'ils sont sous tension, et dans lesquels les données mémorisées n'ont pas à être réécrites périodiquement.

Dispositifs mémoires dynamiques: Dispositifs à semiconducteurs dans lesquels les données mémorisées *ne sont pas* stockées en permanence même s'ils demeurent sous tension, à moins que les données ne soient réécrites périodiquement. Cette dernière opération est appelée *rafraîchissement*.

Mémoire interne Appelée également mémoire *principale* de l'ordinateur. C'est dans cette mémoire que sont conservées les instructions et les données dont a besoin l'UCT pour accomplir sa tâche. C'est la mémoire la plus rapide de l'ordinateur et elle est généralement formée de dispositifs à semiconducteurs.

Mémoire de masse Appelée également mémoire *auxiliaire*. C'est dans cette mémoire que l'on conserve de grandes quantités d'informations en dehors de la mémoire interne de l'ordinateur. Généralement, cette mémoire est plus lente que la mémoire interne et est toujours permanente. Les exemples les plus courants sont le lecteur de disques et le dérouleur de bande.

QUESTIONS DE RÉVISION

1. Définissez les expressions suivantes: a) cellule mémoire; b) mot mémoire; c) adresse; d) octet; e) temps d'accès.
2. La capacité d'une certaine mémoire est 8K x 16. Dites quelle est la longueur en bits de chaque mot. Dites combien on peut y stocker de mots. Indiquez le nombre de cellules mémoires que renferme cette mémoire.
3. Expliquez la différence entre une opération de lecture et une opération d'écriture.
4. *Vrai ou faux:* Une mémoire volatile perd les données qu'elle contient si l'alimentation électrique est interrompue.
5. Expliquez la différence entre une mémoire à accès sélectif et une mémoire à accès séquentiel.
6. Expliquez la différence entre une mémoire VIVE et une mémoire MORTE.
7. *Vrai ou faux:* Une mémoire dynamique retient ses données tant qu'elle est sous tension.

11-2 FONCTIONNEMENT D'UNE MÉMOIRE: GÉNÉRALITÉS

Tous les types de mémoires, quels qu'ils soient et cela malgré leur fonctionnement interne différent, fonctionnent selon de grands principes généraux. Nous croyons qu'une connaissance approfondie de ces principes facilitera l'étude des dispositifs mémoires individuels.

Chaque système mémoire est doté de plusieurs types de lignes d'entrée et de sortie qui ont les rôles suivants:

1. Choisir l'adresse en mémoire qui donne accès à l'emplacement touché par une opération de lecture ou d'écriture.
2. Choisir entre une opération de lecture ou d'écriture.
3. Acheminer la donnée d'entrée mémorisée durant une opération d'écriture.
4. Contenir les données tirées de la mémoire durant une opération de lecture.
5. Valider ou invalider la mémoire afin qu'elle réagisse ou ne réagisse pas aux entrées d'adresse et à la commande lecture/écriture.

La figure 11-3 a) illustre ces fonctions de base sur le schéma simplifié d'une mémoire 32 × 4, soit un dispositif qui mémorise 32 mots de 4 bits. Étant donné que la longueur du mot est de 4 bits, il y a 4 lignes d'entrée de données I_0–I_3 et 4 lignes de sortie de

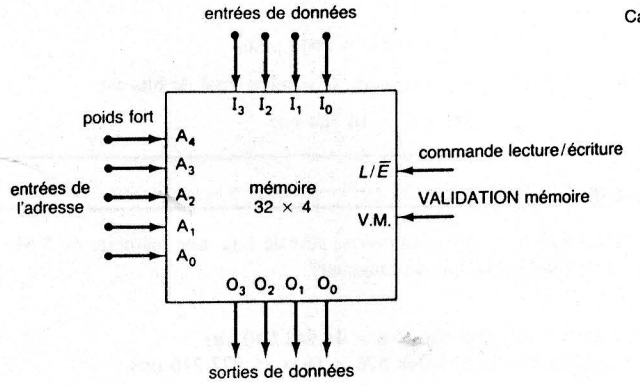


FIGURE 11-3 a) Schéma d'une mémoire 32 × 4; b) organisation virtuelle de cellules mémoires en 32 mots de 4 bits.

données O_0–O_3. Pendant une opération d'écriture, la donnée que l'on place en mémoire est appliquée sur les lignes d'entrée de données. Durant une opération de lecture, le mot prélevé en mémoire apparaît sur les lignes de sortie de données.

Entrées des adresses Il y a dans cette mémoire 32 emplacements de stockage différents, donc 32 adresses différentes allant de 00000_2 à 11111_2 (0 à 31 en décimal). Par conséquent, il faut *cinq* entrées d'adresse, A_0–A_4, pour préciser l'adresse d'un des 32 emplacements. Pour rejoindre un des emplacements en mémoire afin d'effectuer une opération de lecture ou d'écriture, le code d'adresse de 5 bits de cet emplacement particulier doit être placé sur les entrées d'adresse. En général N lignes d'adresse sont nécessaires pour une mémoire de capacité 2^N mots.

Il est possible de concevoir la mémoire de la figure 11-3 a) comme une association de 32 registres, chaque registre pouvant contenir un mot de 4 bits [comme nous le montre la figure 11-3 b)]. Chaque emplacement référencé par une adresse est un groupe de quatre cases mémoires qui renferment des 0 et des 1, dont le regroupement constitue le mot à cette adresse. Par exemple, le mot 0110 est mémorisé à l'adresse 00000, le mot 1001 à l'adresse 00001, et ainsi de suite.

L'entrée $L/\overline{E}$ La ligne d'entrée lecture/écriture ($L/\overline{E}$) précise quelle opération de mémorisation est effectuée. Certains systèmes de mémoire disposent de deux entrées distinctes, une pour la lecture et une pour l'écriture. Quand on utilise une seule entrée $L/\overline{E}$, il est courant que $L/\overline{E} = 1$ désigne une opération de lecture et que $L/\overline{E} = 0$ désigne une opération d'écriture.

La figure 11-4 nous illustre de manière simplifiée les opérations de lecture et d'écriture. D'abord, la partie a) nous montre comment la donnée 0100 est écrite dans le registre mémoire ayant l'adresse 00011. Ce mot est d'abord appliqué sur les lignes de donnée d'entrée de la mémoire, puis il vient remplacer la donnée se trouvant à l'adresse 00011. Par ailleurs, la partie b) nous fait voir comment le mot 1101, se trouvant à l'adresse 11110, est lu. Cette valeur de sortie figure sur les lignes de donnée sortie de la mémoire. Après l'opération de lecture, le mot 1101 est toujours dans la case mémoire 11110. Autrement dit, une opération de lecture ne modifie pas la donnée en mémoire.

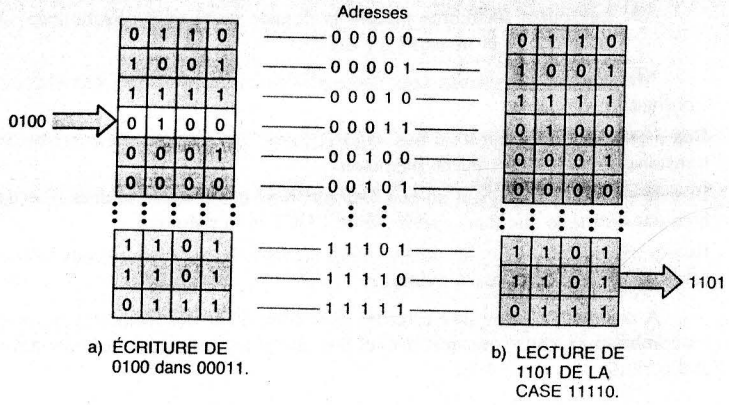


FIGURE 11-4 Illustration simplifiée des opérations de lectures et d'écriture dans une mémoire de 32 × 4. a) Écriture du mot 0100 dans la case mémoire 00011; b) lecture du mot 1101 de la case mémoire 11110.

Validation de la mémoire Un grand nombre de systèmes de mémoire disposent d'un mécanisme d'invalidation de tout le système ou d'une section; quand la mémoire est invalidée, cette dernière ne réagit pas aux actions des autres entrées. C'est le rôle de l'entrée VALIDATION MÉMOIRE du module reproduit à la figure 11-3. Notez que ce nom diffère d'un système de mémoire à un autre. Quand cette entrée est à son niveau vrai HAUT, la mémoire est validée et fonctionne correctement. Au contraire, quand cette entrée est BASSE, la mémoire ne répond plus aux codes d'adresse et à la valeur sur le conducteur $L/\overline{E}$. Une entrée de validation est utile quand on raccorde plusieurs modules de mémoire sur les mêmes bus afin de constituer une mémoire de plus grande capacité. Nous reviendrons sur ce principe plus tard.

EXEMPLE 11-2

Décrivez les conditions de chacune des entrées et des sorties lorsque le contenu de la case dont l'adresse est 00100 doit être lu.

Solution

Entrées d'adresse: 00100
Entrées de donnée: xxxx (inutilisées)
$L/\overline{E}$: HAUT
VALIDATION MÉMOIRE: HAUT
Sorties de donnée: 0001

EXEMPLE 11-3

Décrivez les conditions de chacune des entrées et des sorties quand le mot 1110 doit être écrit dans la case mémoire dont l'adresse est 01101.

Solution

Entrées d'adresse: 01101
Entrées de donnée: 1110
$L/\overline{E}$: BAS
VALIDATION MÉMOIRE: HAUT
Sorties de donnée: xxxx (inutilisées, normalisement dans leur état haute-Z).

EXEMPLE 11-4

Soit une certaine mémoire ayant une capacité de 4K × 8.
a) Dites combien elle a de lignes d'entrée de donnée et de sortie de donnée.
b) Dites combien elle a de lignes d'adresse.
c) Donnez sa capacité en octets.

Solution
a) Huit de chaque, puisque la longueur d'un mot est huit.
b) Cette mémoire stocke 4K = 4 × 1024 = 4096 mots. Il y a donc 4096 adresses mémoires. Puisque $4096 = 2^{12}$, il faut donc que le code d'adresse ait 12 bits pour préciser une adresse parmi 4096.
c) Un octet a 8 bits. Donc la capacité de cette mémoire est de 4096 octets.

La mémoire citée en exemple à la figure 11-3 montre bien les importantes fonctions d'entrée et de sortie communes à la majorité des systèmes mémoires. Évidemment, on retrouvera dans chaque type de mémoire d'autres lignes d'entrée et de sortie qui lui sont propres. D'ailleurs, c'est ce que nous verrons durant l'étude de chacun des types de mémoires.

QUESTIONS DE RÉVISION

1. Dites combien d'entrées d'adresse, d'entrées de donnée et de sorties de donnée il faut pour une mémoire 16K × 12.
2. Quel est le rôle de l'entrée $L/\overline{E}$?
3. Quel est le rôle de la borne VALIDATION MÉMOIRE?

11-3 CONNEXIONS UCT-MÉMOIRE

Une partie importante du présent chapitre est consacrée aux mémoires à semiconducteurs qui servent, comme nous l'avons déjà mentionné, de mémoire interne dans la majorité des ordinateurs modernes. Vous n'avez pas oublié que cette mémoire interne est en communication constante avec l'UCT (l'unité centrale de traitement). Pour comprendre la jonction entre cette mémoire et l'UCT, il n'est pas indispensable de connaître en détail le fonctionnement de l'UCT; c'est pourquoi, nous avons décidé d'exposer succinctement l'interface mémoire-UCT afin que vous ayez une meilleure vue d'ensemble, ce qui facilitera votre compréhension des explications qui suivent.

La mémoire interne d'un ordinateur est constituée de puces de mémoire VIVE et MORTE qui sont reliées à l'UCT par l'intermédiaire de trois faisceaux de conducteurs

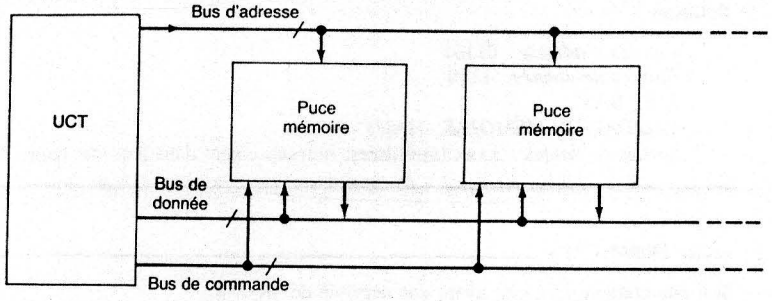


FIGURE 11-5 Trois groupements de conducteurs (bus) réunissent les puces ou barrettes de la mémoire interne à l'UCT.

spéciaux, appelés bus. C'est ce que nous montre la figure 11-5, dans laquelle on peut voir un bus d'adresse, un bus de donnée et un bus de commande. Chacun de ces bus est un assemblage de conducteurs (remarquer la barre oblique, l'indication qu'il y a plus d'un conducteur), et le nombre de conducteurs de chaque bus varie d'un ordinateur à l'autre. Ces trois bus jouent un rôle essentiel dans les processus d'écriture et de lecture de la mémoire par l'UCT.

Quant un ordinateur exécute un programme d'instructions, l'UCT effectue sans arrêt des lectures (prise en charge) de l'information conservée dans les cases mémoires; cette information est soit les codes du programme qui indiquent les opérations à réaliser, ou les données sur lesquelles agissent les opérations. L'UCT mémorise (écrit) des données en mémoire conformément aux ordres qu'elle reçoit. Chaque fois que l'UCT souhaite écrire à un certain endroit en mémoire, les étapes que voici ont lieu:

Opération d'écriture

1. L'UCT place sur le bus d'adresse l'adresse binaire de la case mémoire qu'elle veut rejoindre pour y écrire la donnée.
2. L'UCT place la donnée à écrire sur le bus de donnée.
3. L'UCT active les signaux de commande appropriés qui concourent à réaliser une opération de lecture.
4. Les puces mémoires décodent l'adresse binaire et déterminent la case à laquelle est destinée la donnée.
5. La donnée se trouvant sur le bus de donnée est transmise à la case mémoire sélectionnée.

Chaque fois que l'UCT souhaite lire une donnée à un endroit précis de la mémoire, les étapes suivantes ont lieu:

Opération de lecture

1. L'UCT place sur le bus d'adresse l'adresse binaire de la case mémoire qu'elle veut rejoindre pour y lire une donnée.
2. L'UCT active les signaux de commande appropriés qui concourent à réaliser une opération de lecture.
3. Les puces mémoires décodent l'adresse binaire et déterminent la case dans laquelle se trouve la donnée qui doit être lue.
4. Les puces mémoires placent la donnée de la case sélectionnée sur le bus de donnée, qui la transmet à l'UCT.

Maintenant, vous savez sans doute plus précisément à quoi sert chacun des bus de l'ordinateur.

Bus d'adresse Il s'agit d'un bus *unidirectionnel* qui contient l'adresse binaire, toujours transmise de l'UCT aux puces mémoires.

Bus de donnée Il s'agit d'un bus *bidirectionnel* qui contient la donnée échangée, aussi bien dans un sens que dans l'autre, entre l'UCT et la mémoire.

Bus de commande Ce bus achemine les signaux de commande (comme le signal $R/\overline{W}$) échangés entre l'UCT et les mémoires.

À mesure que nous progresserons dans notre étude des mémoires, nous examinerons les nombreux signaux, transmis par ces bus, et qui accompagnent les opérations de lecture et d'écriture.

QUESTIONS DE RÉVISION

1. Donnez le nom des trois groupements de conducteurs qui mettent l'UCT en communication avec la mémoire.
2. Indiquez sommairement les étapes qui interviennent quand l'UCT lit une donnée en mémoire.
3. Indiquez sommairement les étapes qui interviennent quand l'UCT écrit une donnée en mémoire.

11-4 MÉMOIRES MORTES

Il s'agit ici des mémoires à semiconducteurs qui gardent en permanence des données ou dont les données sont très rarement modifiées. Dans un contexte d'utilisation normale, les données inscrites dans une mémoire MORTE sont inaltérables, elles peuvent seulement être lues. Dans certains types de mémoires MORTES, les données sont introduites durant le procédé de fabrication; dans le cas d'autres mémoires MORTES, les données peuvent être introduites électriquement. On donne le nom de **programmation** ou **de grillage** au processus d'introduction de données dans une mémoire MORTE. Certaines mémoires MORTES ne peuvent jamais avoir d'autres données que celles qui ont été programmées; d'autres toutefois peuvent être **effacées** et reprogrammées aussi souvent qu'on le veut. Plus loin, nous examinerons de façon plus détaillée ces divers types de mémoires MORTES. Pour l'instant, nous allons présumer que les mémoires MORTES ont été programmées et contiennent leurs données.

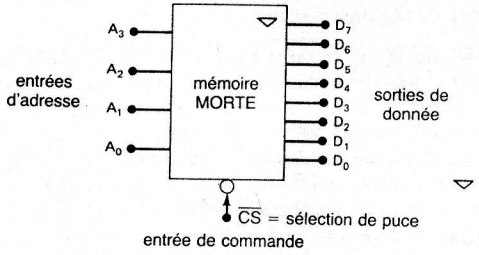
Les mémoires MORTES sont destinées à stocker des données et des informations qui ne doivent pas changer durant le fonctionnement d'un système. Une des principales utilisations des mémoires MORTES est le stockage des programmes dans les micro-ordinateurs. Étant donné que toutes les mémoires MORTES sont *rémanentes*, ces programmes ne sont pas perdus lorsqu'on éteint le micro-ordinateur. Quand le micro-ordinateur est mis sous tension, il peut immédiatement commencer l'exécution du programme mémorisé dans la mémoire MORTE. Ces mémoires servent également à mémoriser le programme et les données d'appareillages commandées par microprocesseur, comme c'est le cas pour les caisses enregistreuses électroniques très sophistiquées, les appareils domestiques et les systèmes de protection.

Schéma fonctionnel d'une mémoire MORTE La figure 11-6 a) reproduit le schéma fonctionnel typique d'une mémoire MORTE. On y voit trois ensembles de signaux: les entrées d'adresse, l'entrée de commande et les sorties de donnée. D'après ce que nous avons dit précédemment, vous pouvez déterminer que cette mémoire MORTE peut contenir 16 mots, puisqu'il y a $2^4 = 16$ adresses possibles et que chaque mot contient 8 bits, puisqu'il y a 8 sorties de données. C'est donc une mémoire MORTE 16 × 8. Une autre façon de décrire la capacité d'une mémoire MORTE est de dire qu'elle stocke 16 octets.

Les sorties de donnée de la plupart des mémoires MORTES intégrées sont des sorties soit à collecteur ouvert ou à trois états, de sorte qu'on peut en raccorder plusieurs au même bus de donnée afin d'étendre la mémoire. Dans la plupart des mémoires MORTES on retrouve 4 ou 8 sorties de donnée, beaucoup plus souvent 8 que 4.

L'entrée de commande $\overline{CS}$ signifie **sélection de puce**. Cette entrée est dans les faits une entrée de validation qui valide ou invalide les sorties de la mémoire MORTE. Certains fabricants désignent autrement cette entrée de commande, par exemple *CE* (validation de puce) ou *OE* (validation de sortie). Beaucoup de mémoires MORTES ont deux entrées de commande, parfois plus, qui doivent devenir actives en même temps pour valider les sorties de donnée. Dans certaines puces de mémoire MORTE, une des entrées de contrôle (généralement *CE*) sert à mettre la mémoire dans un mode d'attente économiseur d'énergie, quand cette dernière n'est pas sollicitée. Cela diminue l'appel de courant sur le bloc d'alimentation.

FIGURE 11-6 a) Schéma fonctionnel d'une mémoire MORTE; b) données programmées; c) données hexadécimales.



a)

Mot	A_3	A_2	A_1	A_0	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
0	0	0	0	0	1	1	0	1	1	1	1	0
1	0	0	0	1	0	0	1	1	1	0	1	0
2	0	0	1	0	1	0	0	0	0	1	0	1
3	0	0	1	1	1	0	1	0	1	1	1	1
4	0	1	0	0	0	0	0	1	1	0	0	1
5	0	1	0	1	0	1	1	1	1	0	1	1
6	0	1	1	0	0	0	0	0	0	0	0	0
7	0	1	1	1	1	1	1	0	1	1	0	1
8	1	0	0	0	0	0	1	1	1	1	0	0
9	1	0	0	1	1	1	1	1	1	1	1	1
10	1	0	1	0	1	0	1	1	1	0	0	0
11	1	0	1	1	1	1	0	0	0	1	1	1
12	1	1	0	0	0	0	1	0	0	1	1	1
13	1	1	0	1	0	1	1	0	1	0	1	0
14	1	1	1	0	1	1	0	1	0	0	1	0
15	1	1	1	1	0	1	0	1	1	0	1	1

b)

Mot	A_3 A_2 A_1 A_0	D_7-D_0
0	0	DE
1	1	3A
2	2	85
3	3	AF
4	4	19
5	5	7B
6	6	00
7	7	ED
8	8	3C
9	9	FF
10	A	B8
11	B	C7
12	C	27
13	D	6A
14	E	D2
15	F	5B

c)

L'entrée $\overline{CS}$ montrée à la figure 11-6 a), est vraie au niveau BAS; par conséquent, elle doit se trouver dans l'état BAS afin d'autoriser la donnée de la mémoire MORTE à apparaître sur les sorties de donnée. Observez l'absence d'une entrée R/W (lecture/écriture), ce qui est logique puisqu'une mémoire MORTE ne peut qu'être lue durant une opération normale.

Opération de LECTURE Supposons que la mémoire MORTE déjà illustrée a été programmée avec les données du tableau de la figure 11-6 b). Seize mots différents sont stockés dans seize emplacements adressables. Par exemple, le mot mémorisé dans l'emplacement désigné par l'adresse 0011 est 10101111. Évidemment, la donnée mémorisée dans la mémoire MORTE l'est sous forme binaire même si souvent on a recours à la notation hexadécimale pour indiquer de façon plus commode la donnée programmée, comme à la figure 11-6 c).

Pour lire un mot de donnée dans une mémoire MORTE, il convient de faire deux choses: appliquer l'adresse où se trouve ce mot sur les entrées d'adresse, puis rendre actives les entrées de commande. Par exemple, si nous voulons lire la donnée mémorisée à l'adresse 0111 de la mémoire MORTE de la figure 11-6, nous devons appliquer sur les entrées d'adresse $A_3A_2A_1A_0 = 0111$ et ensuite appliquer une tension de niveau BAS sur $\overline{CS}$. Les entrées d'adresse sont décodées à l'intérieur de la mémoire MORTE afin de choisir le mot recherché, soit 11101101 qui apparaîtra ainsi sur les sorties D_7–D_0. Si $\overline{CS}$ est gardé au niveau HAUT, les sorties de la mémoire MORTE ne sont pas validées et restent dans leur état haute impédance.

QUESTIONS DE RÉVISION

1. *Vrai ou faux:* Toutes les mémoires MORTES sont rémanentes.
2. Décrivez la marche à suivre pour lire dans une mémoire MORTE.

11-5 ARCHITECTURE DES MÉMOIRES MORTES

L'architecture interne, c'est-à-dire la structure, d'un boîtier de mémoire MORTE est très complexe, et il n'est pas indispensable d'en connaître tous les détails. Toutefois, il peut être instructif d'étudier sommairement un schéma simplifié de l'architecture interne, comme celui de la figure 11-7 décrivant une mémoire MORTE 16 × 8. On y retrouve 4 parties constitutives: *décodeur de rangée, décodeur de colonne, matrice de registres, tampons de sortie*.

Matrice de registres La matrice de registres est l'ensemble des éléments physiques qui mémorisent réellement les données dans la mémoire MORTE. Chaque registre est formé d'un certain nombre de cellules mémoires égal à la longueur du mot. Dans le cas présent, chaque registre contient un mot de 8 bits. Les registres sont agencés selon une matrice carrée comme il est courant dans les puces des mémoires à semiconducteurs. Il est possible d'indiquer la position de chaque registre en donnant sa rangée et sa colonne. Par exemple, le registre 0 se trouve à la rangée 0 et à la colonne 0, alors que le registre 9 se trouve à la rangée 1 et à la colonne 2. Les 8 sorties de données de chaque registre sont connectées à un bus de donnée interne qui court dans tout le circuit. Chaque registre possède deux entrées de validation (E), qui doivent toutes les deux être au niveau HAUT pour que la donnée se trouvant dans le registre adressé soit placée sur le bus.

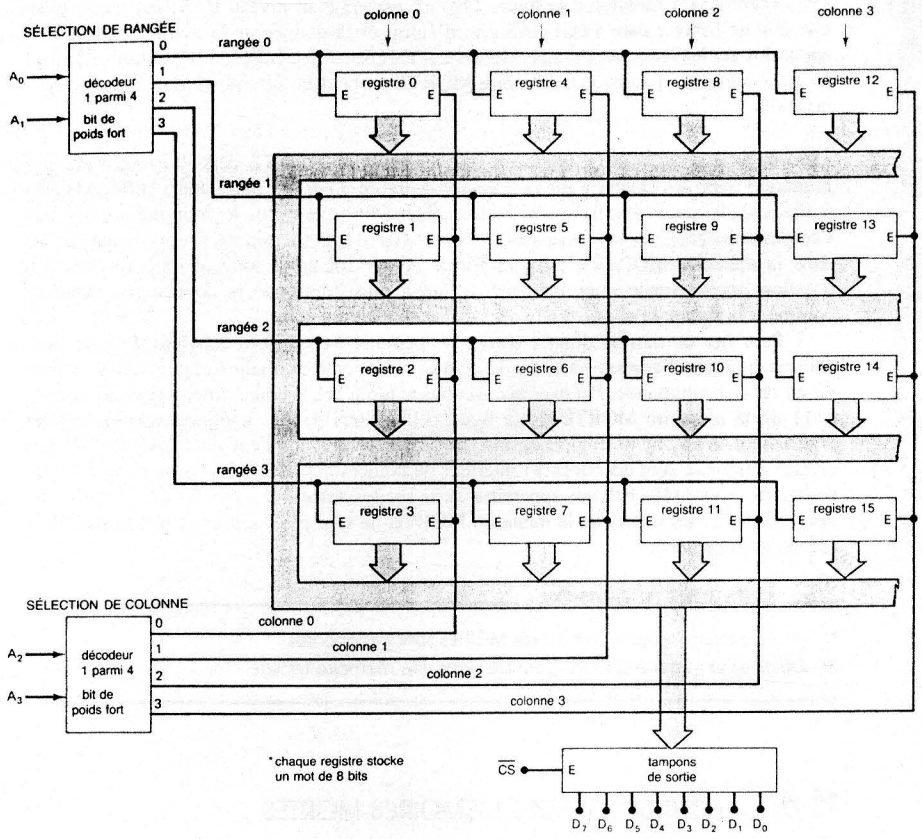


FIGURE 11-7 Architecture d'une mémoire MORTE 16 × 8.

Décodeurs d'adresse L'adresse $A_3A_2A_1A_0$ appliquée sur les entrées détermine le registre de la matrice qui sera validé et dont le mot de 8 bits sera placé sur le bus. Les bits d'adresse A_1A_0 arrivent à un décodeur 1 parmi 4 qui rend active une ligne de sélection de rangée, et les bits d'adresse A_3A_2 arrivent à un second décodeur 1 parmi 4 qui rend active une ligne de sélection de colonne. Comme il n'y aura qu'un registre qui verra sa colonne et sa rangée choisies par les entrées d'adresse, c'est celui-ci qui sera validé.

EXEMPLE 11-5

Dites quel registre est validé par l'adresse 1101.

Solution $A_3A_2 = 11$ ce qui a pour effet de rendre active la ligne de sélection de la colonne 3 du décodeur de colonne, et $A_1A_0 = 01$ ce qui rend active la ligne de sélection de la rangée 1 du décodeur de rangée. Les deux entrées de validation du registre 13 sont alors mises au niveau HAUT, et ce sont les sorties de ce registre qui sont raccordées au bus. Notez que les autres registres de la colonne 3 n'ont qu'une de leur entrée validation qui soit active; de même, pour les registres de la rangée 1.

EXEMPLE 11-6

Quelle adresse valide le registre 7?

Solution Les entrées de validation de ce registre sont connectées respectivement aux lignes de sélection de la rangée 3 et de la colonne 1. Pour choisir la rangée 3, les bits A_1A_0 doivent être à 11, pour choisir la colonne 1, les bits A_3A_2 doivent être à 01. Donc l'adresse recherchée est au complet $A_3A_2A_1A_0 = 0111$.

Tampons de sortie Le registre qui est validé par les entrées d'adresse place sa donnée sur le bus de donnée. Cette donnée vient alimenter des tampons de sortie qui la communiqueront aux sorties de donnée externes si $\overline{CS}$ est au niveau BAS. Si $\overline{CS}$ est au niveau HAUT, les tampons de sortie sont dans leur état haute impédance et D_7–D_0 sont dans l'état flottant.

L'architecture illustrée à la figure 11-7 est analogue à celle que l'on retrouve dans la plupart des boîtiers des mémoires MORTES. Suivant le nombre de mots mémorisés, les registres dans certaines mémoires MORTES ne sont pas agencés selon une matrice carrée. Par exemple, dans le Intel 2708, mémoire MORTE MOS qui stocke 1024 mots de 8 bits, les 1024 registres sont organisés selon une matrice 64 par 16. Dans la pratique, on peut retrouver des mémoires MORTES dont la capacité s'échelonne entre 32 × 8 et 512K × 8.

EXEMPLE 11-7

Décrivez l'architecture interne d'une mémoire MORTE qui stocke 4 K octets et dispose d'une matrice de registres carrée.

Solution 4K signifie en réalité 4 × 1024 = 4096, de sorte qu'il y a dans cette mémoire MORTE 4096 mots de 8 bits. On peut s'imaginer que chaque mot est stocké dans un registre de 8 bits, et qu'il y a 4096 registres branchés sur un bus de donnée commun interne à la puce. Puisque $4096 = 64^2$, les registres sont agencés selon une matrice 64 par 64; c'est-à-dire ayant 64 rangées et 64 colonnes. Il faut donc un décodeur 1 parmi 64 pour décoder les 6 entrées d'adresse correspondant à la sélection de la rangée et un décodeur 1 parmi 64 pour décoder les 6 autres entrées d'adresse correspondant à la sélection de la colonne. Donc il faut 12 entrées d'adresse. Ceci est tout à fait cohérent puisque $2^{12} = 4096$ et qu'il y a au total 4096 adresses différentes.

QUESTIONS DE RÉVISION

1. Dites quelle adresse est nécessaire pour lire la donnée dans le registre 9 de la figure 11-7.
2. Décrivez le rôle du décodeur de sélection de la rangée, du décodeur de sélection de la colonne et des tampons de sortie de la structure d'une mémoire MORTE.

11-6 SYNCHRONISATION DES MÉMOIRES MORTES

Au cours d'une opération de LECTURE, il existe un certain retard entre le moment où on applique une adresse aux entrées de la mémoire MORTE et le moment où une donnée apparaît sur les sorties. Ce retard, appelé **temps d'accès**, t_{ACC}, est une mesure de la vitesse de fonctionnement de la mémoire MORTE. On peut voir sur les chronogrammes de la figure 11-8 une définition du temps d'accès.

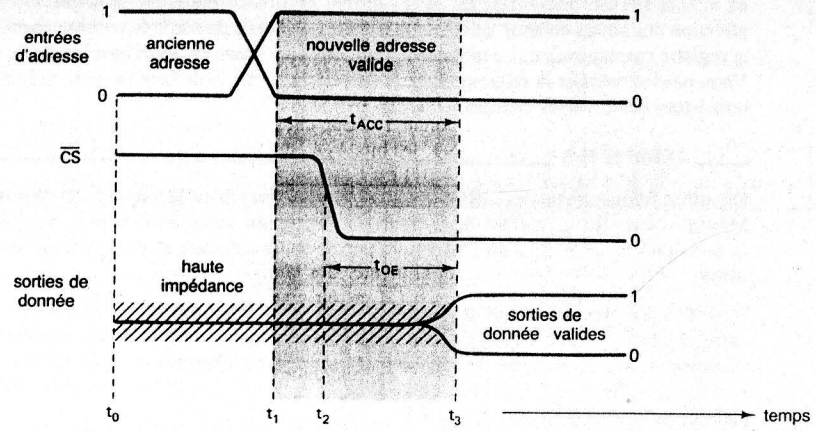


FIGURE 11-8 Synchronisation type d'une opération de LECTURE dans une mémoire MORTE.

La forme d'onde du haut représente les entrées d'adresse, celle du milieu, le signal de sélection de la puce vrai au niveau BAS, $\overline{CS}$, et celle du bas, les sorties de donnée. À l'instant t_0, les entrées d'adresse sont dans un état particulier, certaines sont au niveau HAUT et certaines sont au niveau BAS. De plus, $\overline{CS}$ est au niveau HAUT de sorte que les sorties de donnée de la mémoire MORTE sont dans leur état haute impédance (représenté par des hachures).

Juste avant l'instant t_1, les entrées d'adresse sont modifiées pour refléter la nouvelle adresse désignée par une opération de LECTURE. À t_1, la nouvelle adresse est validée, c'est-à-dire que chaque entrée d'adresse est à son niveau logique valide. À cet instant, les circuits internes de la mémoire MORTE se mettent en marche et décodent les entrées d'adresse pour choisir le registre qui doit envoyer sa donnée au tampon de sortie; à t_2, la sortie $\overline{CS}$ est rendue active pour valider les tampons de sortie. Enfin, à t_3, les sorties passent de leur état haute impédance à l'état correspondant à la donnée mémorisée à l'adresse indiquée.

Le retard entre t_1, le moment où la nouvelle adresse devient valide, et t_3, le moment où les sorties de donnée deviennent valides, est le temps d'accès t_{ACC}. Dans les mémoires MORTES bipolaires types, les temps d'accès se situent dans la plage de 30 à 90 ns, alors que dans les mémoires NMOS ils se situent dans la plage de 35 à 500 ns.

Un autre important paramètre de synchronisation est le *temps de validation de sortie*, t_{OE}, qui est le retard entre le moment où l'entrée $\overline{CS}$ passe au niveau vrai et le moment où la donnée paraît sur les sorties. Les valeurs typiques de t_{OE} sont 20 ns pour les mémoires MORTES bipolaires et 26 à 100 ns pour les mémoires NMOS. Ce paramètre de synchronisation est une caractéristique importante à prendre en considération dans les cas où les nouvelles valeurs figurent déjà sur les entrées d'adresse, pendant que les sorties de la mémoire MORTE ne sont pas encore validées. Lorsque $\overline{CS}$ passe au niveau BAS pour valider les sorties, le retard qu'on observera est t_{OE}.

11-7 TYPES DE MÉMOIRES MORTES

Maintenant que vous comprenez de façon générale l'architecture interne et le fonctionnement externe d'une mémoire MORTE, abordons brièvement les divers types de mémoires MORTES afin de constater les différences existant dans la façon de les programmer et d'examiner leur possibilité d'être effacées et reprogrammées. Dans la partie qui suit vous constaterez que la plupart des types de mémoires MORTES sont désignés au moyen d'un sigle créé en utilisant comme racine le sigle anglais de mémoire MORTE, soit ROM.

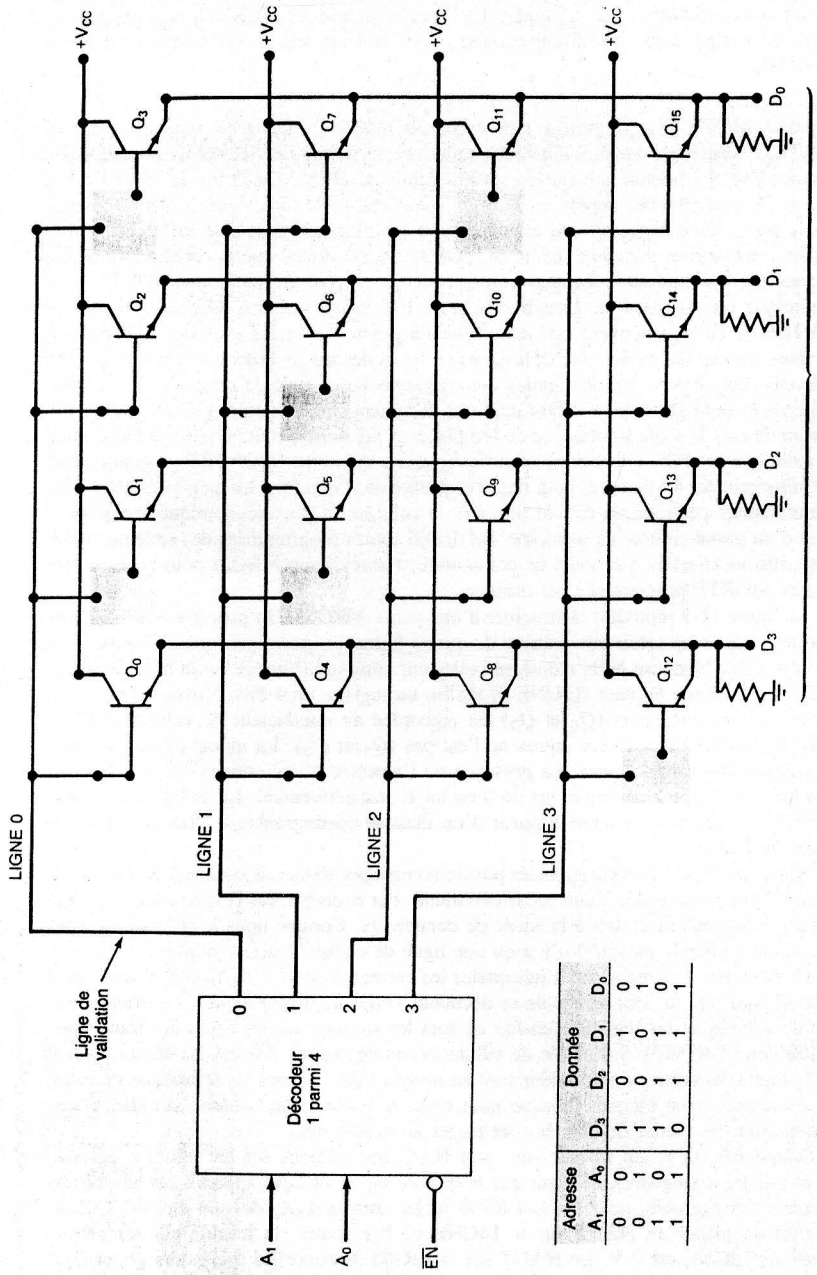
Mémoire MORTE programmée par masque (MROM) Dans ce type de mémoire MORTE, l'écriture de données dans les emplacements mémoires (leur programmation) est effectuée par le fabricant suivant les spécifications du client. On utilise le négatif d'une épreuve photographique, appelé un *masque*, pour établir les interconnexions électriques dans la puce. Un masque spécial est nécessaire pour chaque ensemble différent d'informations à mémoriser dans une mémoire MORTE. Étant donné que la constitution de ces masques est une opération coûteuse, le recours à ce type de mémoire MORTE n'est économique que si vous en avez besoin d'un très grand volume. Certaines mémoires MORTES de ce type peuvent être achetées déjà préprogrammées avec des informations courantes comme des tables mathématiques et des codes de génération de caractères pour les écrans cathodiques. Un des grands inconvénients de ce type de mémoire est l'impossibilité de la reprogrammer s'il faut apporter des changements au programme mémorisé. Dans un tel cas, la seule solution est de remplacer la mémoire MORTE par une autre ayant le programme modifié. De nombreux autres types de mémoires MORTES programmables par l'utilisateur ont été élaborés pour régler ce problème. Toutefois, les mémoires MORTES programmables par masque restent toujours la solution la plus économique lorsqu'on a besoin d'un grand volume de mémoires MORTES toutes programmées de la même façon. Nous utilisons ci-après, par souci de commodité, l'abréviation MROM pour parler d'une mémoire MORTE programmée par masque.

La figure 11-9 reproduit la structure d'une petite MROM. On peut y voir 16 cellules mémoires organisées selon une matrice de quatre lignes et quatre colonnes. Chaque case est un transistor bipolaire NPN monté en collecteur commun (l'entrée est la base, la sortie l'émetteur). La ligne du haut (LIGNE 0) réalise un registre de 4 bits. Voyez comment la base de certains transistors (Q_0 et Q_2) est raccordée au conducteur de validation de la LIGNE 0, tandis que celle des autres ne l'est pas (Q_1 et Q_3). La même chose est vraie pour chacune des autres lignes. La présence ou l'absence d'une connexion avec la base détermine si la valeur mémorisée est un 0 ou un 1, respectivement. La connexion ou non de la base est réalisée en usine à partir d'un masque photographique élaboré selon les données du client.

Notez que les cellules occupant les positions correspondantes de chaque ligne (registre) ont leur émetteur raccordée à une sortie commune. Par exemple, les émetteurs de Q_0, Q_4, Q_8 et Q_{12} sont tous raccordés à la sortie de donnée D_3. Comme nous le verrons, cela ne cause aucun problème puisqu'il n'y a qu'une ligne de cellules qui est validée à la fois.

Le décodeur 1 parmi 4 sert à interpréter les entrées d'adresse A_1A_0 et à trouver quel registre (ligne) sera lu. Les sorties de ce décodeur vraies au niveau HAUT constituent les lignes de validation des registres, en fait ce sont les signaux sur les bases des transistors des différentes LIGNES. Si l'entrée de validation du décodeur, $\overline{EN}$, est gardée au niveau HAUT, toutes les sorties du décodeur sont au niveau BAS, et tous les transistors de cette matrice mémoire sont bloqués puisque pour tous, la tension sur la base est nulle. Dans cette situation, les sorties de donnée sont toutes au niveau bas.

Quand $\overline{EN}$ est à son niveau vrai, soit BAS, les données sur les lignes d'adresse valident une ligne (registre), de sorte que la donnée qui s'y trouve apparaît sur les sorties de donnée. Par exemple, pour lire la LIGNE 0, les entrées A_1A_0 doivent être 00. Cela a pour effet de placer un HAUT sur la LIGNE 0. Par contre, la tension sur les autres conducteurs LIGNE est 0 V. Le HAUT sur la LIGNE 0 ouvre les transistors Q_0 et Q_2, mais pas Q_1 et Q_3. Puisque Q_0 et Q_2 conduisent, les sorties de la donnée D_3 et D_1 sont



Adresse		Donnée			
A_1	A_0	D_3	D_2	D_1	D_0
0	0	1	0	1	0
0	1	1	0	0	1
1	0	1	1	1	0
1	1	1	1	1	1

FIGURE 11-9 Structure d'une MROM bipolaire; chaque cellule renferme un transistor bipolaire. Une base non raccordée mémorise un « 0 », et une base raccordée, un « 1 ».

au niveau HAUT: les sorties D_2 et D_0 restent au niveau BAS. De la même façon, l'application des autres codes d'adresse fournit sur les bornes de sortie le nombre contenu dans le registre correspondant. Le tableau de la figure 11-9 passe en revue tous les cas possibles. Vous pouvez vérifier la relation entre la donnée et le raccordement ou non de la base des transistors des diverses cellules.

EXEMPLE 11-8

On utilise fréquemment des MROM pour mémoriser des tables de fonctions mathématiques. Montrez comment la MROM de la figure 11-9 pourrait servir à mémoriser les valeurs de la fonction $y = x^2 + 3$; x est l'adresse d'entrée de la mémoire et y, la donnée fournie en sortie.

Solution La première étape est la constitution de la table qui établit la correspondance entre la sortie voulue et les entrées. Le nombre binaire entrée, x, est représenté par l'adresse A_1A_0. Le nombre binaire sorti est la valeur recherchée y. Par exemple, quand $x = A_1A_0 = 10_2 = 2_{10}$, la sortie doit valoir $2^2 + 3 = 7_{10} = 0111_2$. Voici la table complète de cette fonction pour $x = 0$ à 3.

x		$y = x^2 + 3$			
A_1	A_0	D_3	D_2	D_1	D_0
0	0	0	0	1	1
0	1	0	1	0	0
1	0	0	1	1	1
1	1	1	1	0	0

Cette table est envoyée au fabricant de la mémoire morte programmable qui réalisera à partir d'elle un masque qui contrôlera le raccordement ou non des bases des transistors durant le processus de fabrication. Par exemple, la première ligne de la table indique que les bases de Q_0 et Q_1 ne sont pas raccordées, alors que celles de Q_2 et Q_3 le sont.

Les MROM bipolaires sont offertes sur le marché en différentes capacités. Une de ces mémoires les plus répandues est la 74187. C'est une mémoire 256 × 4, dont le temps d'accès est de 40 ns. Ses sorties sont de type à collecteur ouvert, ce qui nécessite l'installation à l'extérieur de résistances de rappel à la source. Une autre mémoire MORTE bipolaire est la 7488A, dont la capacité est de 32 × 8 et dont le temps d'accès est 45 ns.

Les MROM de type MOS ont une structure interne semblable à celle de la figure 11-9, la seule différence étant l'utilisation de transistors MOS au lieu de transistors bipolaires. Le TMS47256 est une mémoire à masque NMOS dont la capacité est 32K × 8. On peut voir son symbole à la figure 11-10. Observez les sorties trois états qui autorisent un interfaçage avec un bus de donnée d'ordinateur. En plus des 14 entrées d'adresse, cette puce renferme deux sorties de validation: $\overline{E}$ et $\overline{S}$. Ces deux bornes doivent être au niveau BAS pour valider les sorties de la MROM. L'entrée $\overline{E}$ est en outre associée à une fonction **réduction de la consommation**. Quand $\overline{E}$ est gardée au niveau HAUT, les circuits internes de la puce sont dans un mode d'attente à faible consommation d'énergie, qui ne prélève que le quart du courant d'alimentation habituel. Le TMS47256 a un temps d'accès de 200 ns et une consommation d'attente de 82,5 mW. La version CMOS, le TMS47C256 a un temps d'accès de 150 ns et une consommation d'attente de 2,8 mW.

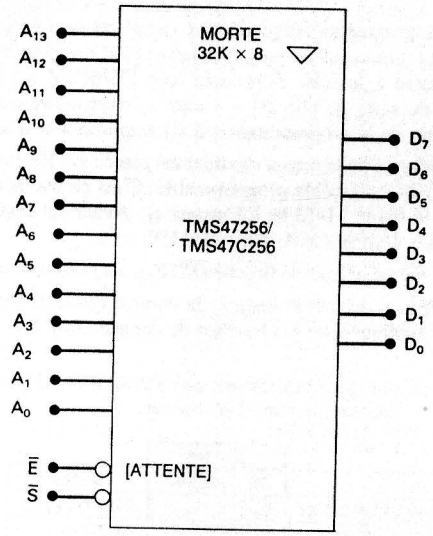


FIGURE 11-10 Symbole logique de la MROM TMS4256 fabriquée selon la technologie CMOS/NMOS.

Mémoire MORTE programmable (PROM) Une mémoire MORTE programmée par un masque est un élément très coûteux qui ne convient qu'à des applications de très grande série, dans lesquelles le coût est amorti sur un très grand nombre d'unités. Lorsqu'il s'agit d'applications de volume plus modeste, les fabricants ont développé des PROM avec des **liaisons fusibles**, que l'utilisateur peut programmer; cela veut donc dire que la programmation n'est plus faite en usine mais est faite par l'utilisateur selon ses besoins. Cependant, une PROM une fois programmée ne diffère en rien d'une MROM et elle ne peut être ni effacée ni reprogrammée. Donc si une PROM se révèle erronée, il faut la jeter et en utiliser une autre.

La structure à liaisons fusibles d'une PROM ressemble beaucoup à celle d'une MROM puisqu'on y retrouve certaines cellules mémoires dont les connexions sont ouvertes et d'autres dont les connexions sont fermées pour représenter un 1 ou un 0. Dans la MROM de la figure 11-9, ces connexions unissent les lignes de validation et les bases des transistors. Dans une PROM, chacune de ces connexions est un petit fil fusible mis là par le fabricant (figure 11-11). L'utilisateur peut choisir de *griller* n'importe laquelle de ces liaisons fusibles pour enregistrer en mémoire les données voulues. Habituellement, on fait fondre les petites connexions en appliquant au dispositif une tension bien précise qui produit l'intensité de courant nécessaire pour faire fondre la petite connexion qui s'ouvre alors comme lorsqu'un fusible grille. Dès que le filament est fondu, il est impossible de revenir en arrière.

Le processus de programmation d'une PROM et de la vérification des données programmées est excessivement long et laborieux lorsqu'il est effectué manuellement. Pour une somme assez modeste on peut acheter sur le marché des programmateurs de PROM qui permettent d'introduire à partir d'un clavier le programme dans une mémoire VIVE, et qui, par la suite, se chargent de faire griller les liaisons et de vérifier automatiquement le programme introduit sans l'intervention de l'utilisateur.

Une PROM intégrée bipolaire assez répandue est la 74186, dont la structure est 64 mots de 8 bits; son temps d'accès est de 50 ns. Une autre est la TBP28S166 dont la

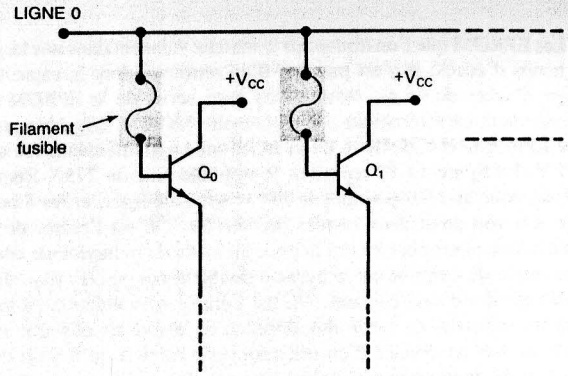


FIGURE 11-11 PROM réalisée avec des filaments fusibles que l'utilisateur a le loisir de faire fondre ou non selon les besoins de son programme.

capacité est 2K × 8. Les PROM MOS qui sont offertes dans le commerce ont des capacités qui dépassent de beaucoup celles des mémoires MORTES programmables bipolaires. Ainsi, la TMS27PC256 est une PROM CMOS dont la capacité est de 32K × 8, et la consommation d'attente, de 1,4 mW. Les temps d'accès varient entre 120 et 250 ns.

Mémoire MORTE effaçable programmable (EPROM) Une EPROM peut être programmée par un utilisateur, mais il est en outre possible de l'*effacer* et de la reprogrammer aussi souvent qu'on le désire. Une EPROM programmée est une mémoire *rémanente* qui retient ses données indéfiniment. La programmation d'une EPROM fait intervenir l'application de niveaux de tension spéciaux (typiquement dans la gamme de 10 à 25 V) sur des entrées précises de la puce pendant une durée spécifiée (habituellement 50 ns par emplacement adressable). Cette programmation est généralement l'oeuvre d'un circuit programmateur spécial étranger au circuit dans lequel l'EPROM sera insérée. La programmation complète d'une puce EPROM peut prendre jusqu'à quelques minutes.

Les cellules de stockage d'une EPROM sont des transistors à effet de champ munis d'une grille de silicium sans connexion électrique (c'est-à-dire une *grille flottante*). Dans son état normal, chaque transistor est bloqué, et la cellule contient un 1. En appliquant au dispositif une impulsion de programmation haute tension spéciale, des électrons très énergétiques sont injectés dans la région de la grille flottante ce qui « ouvre » le transistor. Les électrons demeurent piégés dans la région quand cesse l'impulsion, puisqu'il n'y a pas de trajet de décharge. Le transistor reste en conduction même après le retrait de la tension; la cellule mémorise alors un zéro. Durant ce processus de programmation, on utilise les broches d'adresse et de donnée de la puce pour diriger l'impulsion de programmation vers les cellules visées.

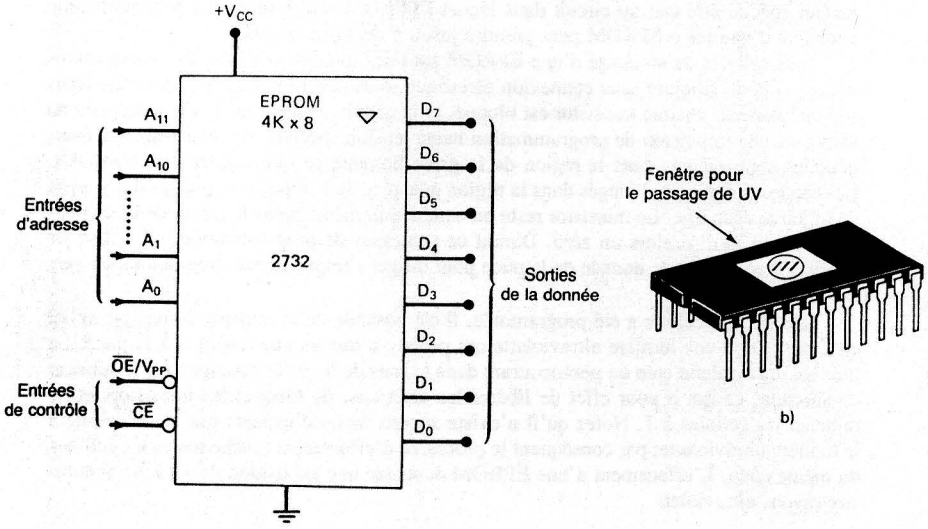
Quand une cellule a été programmée, il est possible de la ramener à son état initial en l'exposant à une lumière ultraviolette qui pénètre à travers une fenêtre sur la puce. La lumière ultraviolette crée un photocourant dans le sens de la grille flottante vers le substrat de silicium, ce qui a pour effet de libérer les électrons, de bloquer les transistors et de ramener les cellules à 1. Notez qu'il n'existe aucune façon d'exposer une seule cellule à la lumière ultraviolette; par conséquent le processus d'effacement touche toutes les cellules du même coup. L'effacement d'une EPROM demande une exposition de 15 à 30 minutes aux rayons ultraviolets.

Les EPROM que l'on trouve sur le marché viennent dans une large gamme de capacités et de temps d'accès: il n'est pas rare d'en retrouver dont la capacité est de 128K × 8 et le temps d'accès de 45 ns. Nous allons nous servir de la EPROM 2732 pour illustrer le fonctionnement caractéristique d'une mémoire MORTE effaçable programmable. La 2732 est une EPROM NMOS 4K × 8 qui fonctionne normalement avec une seule alimentation de +5 V. La figure 11-12 reproduit le symbole bloc du 2732. Remarquez les 12 entrées d'adresse, pour un total possible de $2^{12} = 4096$ adresses, et les 8 bornes de sortie pour la donnée. On voit aussi deux entrées de contrôle. $\overline{CE}$ est l'entrée de validation de la puce que l'on utilise aussi pour mettre la puce en mode de réduction de consommation (attente). L'autre entrée de contrôle est la borne à double fonction $\overline{OE}/V_{PP}$, dont le rôle varie selon le mode opératoire de l'élément. $\overline{OE}$ est l'entrée de validation de la sortie et sert à commander les tampons de sortie des données du dispositif afin que ce dernier puisse être connecté au bus de donnée d'un microprocesseur, sans qu'il y ait de conflit de bus. V_{PP} est une tension de programmation spéciale que l'on applique quand on programme la puce.

Le 2732 a plusieurs modes opératoires qui sont activés selon la combinaison des tension sur $\overline{CE}$ et sur $\overline{OE}/V_{PP}$; voyez le tableau de ces modes à la figure 11-13 a). Le mode de lecture et le mode d'invalidation de la sortie sont les deux modes de fonctionnement normaux de l'EPROM quand elle est installée dans le système de mémoire d'un ordinateur et qu'elle doit être lue fréquemment par l'UCT. Pour ces modes, $\overline{CE}$ est à son niveau vrai BAS, et le niveau imposé à $\overline{OE}$ détermine si oui ou non les broches de sortie de l'élément contiennent une donnée.

Le mode d'attente ou réduction de la consommation est activé en mettant $\overline{CE}$ au niveau inactif HAUT. Ce mode est employé pour mettre le dispositif en attente quand l'UCT n'y accède pas. L'EPROM est sourde à tous les autres signaux d'entrée quand elle est en attente. Dans le mode d'attente, la consommation est de 175 mW, contre 500 mW dans les autres modes.

FIGURE 11-12 a) Symbole logique de l'EPROM 2732; b) boîtier type d'une EPROM avec la fenêtre pour le passage des rayons d'effacement UV.



Le mode programmation est activé quand on veut écrire de nouvelles données dans la mémoire. Le processus de programmation peut inscrire des mots de 8 bits à un emplacement d'adresse à la fois. Supposons que l'EPROM a été effacée par de la lumière ultraviolette, de sorte qu'elle est « vierge », c'est-à-dire qu'elle ne contient que des 1. Voici les étapes de la programmation d'un emplacement d'adresse quelconque:

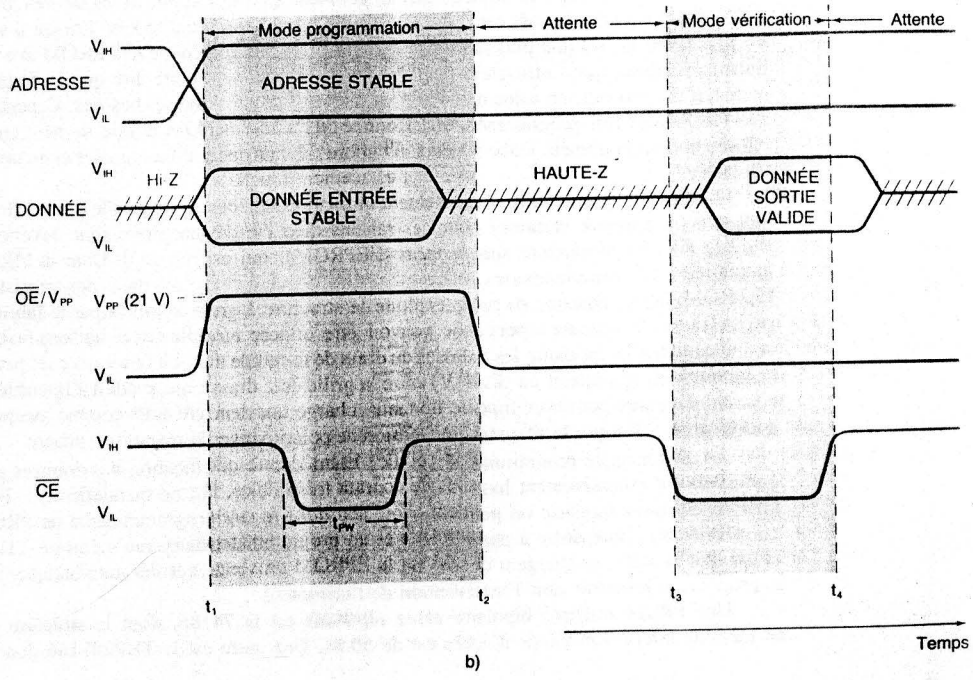
1. L'adresse de la case à modifier est placée sur les entrées d'adresse de la mémoire MORTE effaçable programmable. C'est ce que nous fait voir le chronogramme de la figure 11-13 b) à l'instant t_1. Avant cet instant, l'élément était dans son mode d'attente avec $\overline{CE}$ = HAUT.
2. Au même instant, la broche $\overline{OE}/V_{PP}$ est portée au niveau de tension V_{PP} (21 V).
3. Toujours au même instant, la donnée qui doit être inscrite à l'adresse fournie est appliquée sur les broches de donnée D_7 à D_0. En mode de programmation,

FIGURE 11-13 a) Modes opératoires de l'EPROM 2732; b) chronogrammes des modes programmation et vérification.

Mode	Entrées $\overline{CE}$	Entrées $\overline{OE}/V_{PP}$	Sorties
LECTURE VÉRIFICATION	V_{IL}	V_{IL}	$DONNÉE_{SORTIE}$
INVALIDATION SORTIE	V_{IL}	V_{IH}	HAUTE-Z
ATTENTE	V_{IH}	X	HAUTE-Z
PROGRAMMATION	V_{IL}	V_{PP}	$DONNÉE_{ENTRÉE}$

Note: V_{IL} = BAS TTL
V_{IH} = HAUT TTL
X = indifférent
V_{PP} = nominal 21 V

a)



ces broches de donnée agissent comme des *entrées* parce que la forte tension sur la broche $\overline{OE}/V_{PP}$ désactive les tampons de sortie de la donnée du dispositif.

4. Une fois que l'adresse et la donnée sont stabilisées, l'entrée $\overline{CE}$ est ramenée au niveau BAS pendant une durée caractéristique de $t_{PW} = 50$ ms. Durant cet intervalle, chaque mémoire particulière de l'adresse sélectionnée reste à 1 ou passe à 0 selon le bit correspondant de la donnée à inscrire.
5. Quand le processus de programmation est terminé, l'entrée $\overline{CE}$ revient au niveau HAUT, et la borne $\overline{OE}/V_{PP}$ est remise au niveau BAS. Dans le mode d'attente, les entrées de donnée sont déconnectées des broches de donnée de la mémoire qui sont portées dans leur état haute impédance.
6. La dernière étape de la programmation est la vérification que le mot envoyé se trouve bel et bien à l'adresse spécifiée. Le mode de vérification est en réalité le mode de lecture. L'entrée $\overline{OE}$ est gardée au niveau BAS et $\overline{CE}$ est portée momentanément au niveau BAS à l'instant t_3. Cela commande aux circuits internes de l'EPROM d'envoyer la donnée se trouvant à l'adresse indiquée dans les tampons de sortie, pour qu'elle puisse être lue.

Le processus de programmation peut être effectué manuellement, ce qui alors prend quelques heures pour une puce 2732. Par contre, il existe dans le commerce des programmateurs d'EPROM qui permettent de programmer et de vérifier la totalité d'un 2732 en moins de deux minutes, après que l'utilisateur a entré au clavier ses données dans la mémoire VIVE du programmateur. On peut remplir la mémoire VIVE de certains programmateurs en transférant directement les données se trouvant dans un ordinateur. Cette procédure, appelée *chargement*, permet à l'utilisateur de développer d'abord son programme à l'ordinateur, ensuite, quand tout est parfait, de transférer les données de la mémoire de l'ordinateur dans la mémoire du programmateur. L'EPROM peut ensuite être configurée dans le programmateur.

PROM effaçable électriquement (EEPROM) Comme on l'a déjà mentionné, les EPROM sont affectées par deux inconvénients majeurs. D'abord, il faut les retirer de leur fiche pour les effacer et les reprogrammer. Ensuite, l'effacement touche tout le contenu de la mémoire; il faut donc reprogrammer complètement la mémoire même si nos changements ne touchent que quelques mots. C'est pour ces raisons qu'a été développée autour des années 1980 la *PROM effaçable électriquement* (EEPROM), qui est dans les faits une version améliorée de l'EPROM.

L'EEPROM, tout comme l'EPROM, exploite la structure à grille flottante dont on a déjà parlé. Toutefois, elle a la faculté de pouvoir être effacée *électriquement* grâce à l'addition d'une mince région d'oxyde au-dessus du drain de la cellule mémoire du transistor MOS. En appliquant une haute tension (21 V) entre la grille et le drain du transistor MOS, il y a induction d'une charge dans la grille flottante, charge qui demeure là même après la mise hors tension. L'application de la même tension de polarité inverse libère les charges piégées dans la grille flottante et efface la cellule. Étant donné que ce mécanisme de transport de charge se fait avec des intensités de courant très faibles, la programmation et l'effaçage d'une EEPROM peut généralement avoir lieu *dans le circuit* (c'est-à-dire sans devoir utiliser une source de lumière ultraviolette et un programmateur de PROM).

La principale supériorité de la EEPROM sur la EPROM provient de sa faculté d'être effaçable électriquement et de permettre la reprogrammation de mots *individuels* de la matrice mémoire. Un autre point avantageux de la EEPROM est la possibilité de l'effacer en environ 10 ms (dans le circuit) en regard d'environ 30 minutes pour une EPROM soumise à une lumière ultraviolette extérieure. En outre, une EEPROM est programmée

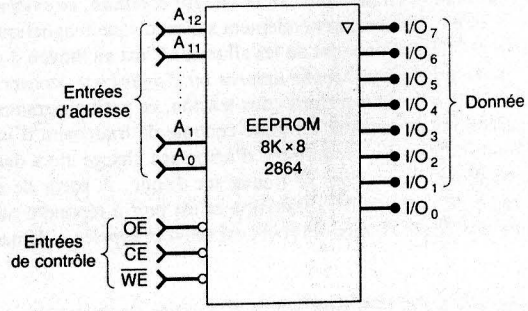
plus rapidement: il faut appliquer une impulsion de programmation de 10 ms seulement pour chaque mot de donnée, alors que pour la EPROM, cette impulsion doit durer 50 ms.

Comme on l'a dit, la EEPROM peut être effacée et reprogrammée par le biais de l'application de tensions appropriées, ce qui indique qu'on n'a pas à la retirer du circuit où elle se trouve si certains composants supplémentaires font déjà partie intégrante des circuits. Les composants supplémentaires dont on veut parler sont une tension de programmation de 21 V (V_{PP}), généralement créée à partir de l'alimentation de +5 V au moyen d'un convertisseur de tension continue, et des circuits pour commander le signal de synchronisation de 10 ms et l'ordonnancement des opérations d'effaçage et de programmation.

L'Intel 2816 a été la première EEPROM mise sur le marché en 1981. Elle avait une capacité de 2K × 8, un temps d'accès de 250 ns et les caractéristiques décrites ci-dessus. Depuis lors, les percées dans la conception des EEPROM ont produit des dispositifs nettement améliorés. Le 2864 est une EEPROM de 8K × 8 qui renferme dans son boîtier tous les circuits nécessaires pour générer les fortes tensions qui rendent possible l'effacement et la reprogrammation, de sorte qu'il ne faut dans cette puce qu'une broche d'alimentation $+V_{CC}$. Pour cette raison, le 2864 est un élément de mémoire presque aussi facile à utiliser qu'un module de mémoire VIVE dont nous parlerons plus loin. Évidemment, l'EEPROM, contrairement à la mémoire VIVE, n'est pas volatile et conserve les données qu'on y a inscrites même après la mise hors tension. En revanche, les mémoires VIVES statiques ont des circuits internes moins complexes et des temps d'accès beaucoup plus courts.

La figure 11-14 reproduit en a) le schéma de l'EEPROM 2864; en b), on peut voir le tableau des différents modes opératoires de cette mémoire. Observez l'utilisation du sigle I/O pour la désignation des broches de donnée, puisque ces broches fonctionnent autant comme entrées que comme sorties, suivant le mode activé. Trois entrées de contrôle fixent le mode opératoire. $\overline{CE}$ a pour rôle de valider ou d'invalider la puce; quand elle est invalidée, la puce se trouve dans le mode d'attente et alors, elle consomme moins d'énergie. $\overline{OE}$ sert à valider ou à invalider les tampons de sortie. En mode de lecture, $\overline{OE} = 0$, ce qui valide les tampons de sortie; durant une opération de lecture, $\overline{OE} = 1$ et les tampons de sortie sont invalidés; il est désormais possible de placer sur les broches I/O une donnée d'entrée. En mode d'écriture, $\overline{WE}$ est amenée au niveau BAS pendant que les données à écrire sont placées sur les broches I/O. Les circuits internes effacent automatiquement, avant l'écriture, les cellules se trouvant à l'emplacement adressé.

FIGURE 11-14 Schéma de l'EEPROM 2864; b) modes opératoires.



a)

	Entrées			
Mode	$\overline{CE}$	$\overline{OE}$	$\overline{WE}$	Sorties
LECTURE	V_{IL}	V_{IL}	V_{IH}	$DONNÉE_{SORTIE}$
ÉCRITURE	V_{IL}	V_{IH}	V_{IL}	$DONNÉE_{ENTRÉE}$
ATTENTE	V_{IH}	X	X	HAUTE-Z

b)

QUESTIONS DE RÉVISION

1. *Vrai ou faux:* Une MROM peut être programmée par l'utilisateur.
2. Indiquez la différence entre une PROM et une MROM. Sont-elles effaçables et reprogrammables?
3. *Vrai ou faux:* Les PROM existent en versions bipolaire et MOS.
4. Indiquez comment une EPROM peut être effacée.
5. *Vrai ou faux:* Il est impossible d'effacer seulement une partie d'une mémoire EPROM.
6. Décrivez brièvement le déroulement de la programmation et de la vérification d'une EPROM 2732.
7. Dites quels sont les avantages d'une EEPROM par rapport à une EPROM.

11-8 APPLICATIONS DES MÉMOIRES MORTES

Les mémoires MORTES sont utilisées partout où il faut garder en permanence des données qu'on ne change que très rarement, sinon jamais. Nous décrivons ici quelques-unes des applications les plus courantes des mémoires MORTES.

Stockage de programmes de micro-ordinateur (microprogrammation) À l'heure actuelle, c'est l'application la plus universelle des mémoires MORTES. Les micro-ordinateurs personnels et de gestion utilisent des mémoires MORTES pour stocker ou emmagasiner les programmes du système d'exploitation et les interpréteurs de langage (tel BASIC) qui rendent l'ordinateur immédiatement utilisable dès qu'on le met sous tension. De plus, les produits commandés par micro-ordinateur ont recours à des mémoires MORTES pour emmagasiner les programmes de commande. Comme exemple de dispositifs commandés par micro-ordinateur, mentionnons les jeux électroniques, les caisses enregistreuses électroniques, les balances électroniques et le système d'injection de carburant des automobiles.

On donne souvent le nom de *microprogrammation* à l'installation des programmes dans la mémoire MORTE d'un micro-ordinateur, parce que ces programmes ne sont pas appelés à être changés. Par ailleurs, les programmes qui sont placés dans les mémoires VIVES sont appelés *des logiciels* parce qu'on peut les modifier facilement.

Mémoire d'amorçage Nombre de micro-ordinateurs et la majorité des grands ordinateurs n'ont pas leur système d'exploitation en mémoire MORTE. Au contraire, ce système se trouve dans une mémoire de masse extérieure, généralement sur un disque magnétique. Comment alors ces machines savent-elles qu'on vient de les allumer? C'est au moyen d'un petit programme relativement court, appelé *programme amorce ou d'amorçage*, conservé dans une mémoire MORTE. Quand l'ordinateur est mis sous tension, ce petit programme s'exécute de lui-même. Ses instructions demandent à l'unité centrale de traitement d'initialiser les ressources matérielles installées. Le programme d'amorçage charge alors dans la mémoire principale le système d'exploitation qui se trouve sur disque. À partir de ce point, l'ordinateur est sous la régie du système d'exploitation et est prêt à répondre aux commandes de l'utilisateur. Cette procédure de mise en route est souvent appelée « démarrage du système ».

Tables de données Les mémoires MORTES sont souvent utilisées pour stocker des tables de données qui ne changent pas. Par exemple, des tables trigonométriques (sinus, cosinus) et des tables de conversion de code.

De nombreuses tables de consultation réalisées en mémoire MORTE existent pour les fonctions trigonométriques. L'une d'elles, la MM4220BM de National Semiconductor, mémorise la fonction sinus pour les angles entre 0 et 90 degrés. Cette mémoire MORTE a une structure de 128 × 8, sept entrées d'adresse et huit sorties de donnée. L'angle est introduit sur les entrées d'adresse par pas d'accroissement d'environ 0,7°. Par exemple, l'adresse 0000000 est 0°, l'adresse 0000001 est 0,7°, l'adresse 0000010 est 1,41° et ainsi de suite jusqu'à l'adresse 1111111, qui correspond à 89,3°. Lorsqu'une adresse est appliquée à la mémoire MORTE, la donnée qui apparaît sur les sorties représente approximativement le sinus de l'angle. Par exemple, quand l'adresse introduite est 1000000 (représentant approximativement 45°), les sorties de donnée affichent 10101101. Étant donné que le sinus est toujours inférieur ou égal à 1, ces données sont interprétées comme une fraction, c'est-à-dire .10110101, ce qui correspond après conversion en décimal à 0,707 (la valeur du sinus de 45°).

Convertisseur de données Un *convertisseur de données* est un circuit dont l'entrée est une donnée exprimée dans un certain type de code et la sortie, la même donnée exprimée dans un autre type de code. Il est nécessaire d'effectuer des conversions de code, par exemple, quand un ordinateur fournit ses données de sortie en code binaire pur et qu'on veut les avoir en code DCB pour piloter des afficheurs DEL à 7 segments.

L'une des méthodes les plus simples pour convertir un code est l'utilisation d'une mémoire MORTE programmée de façon à ce que l'application d'une adresse particulière (le code de départ) produise en sortie l'équivalent de la donnée exprimée dans un nouveau code. Le boîtier 74185 est une mémoire MORTE TTL dans laquelle est programmée la conversion binaire-DCB pour un nombre de 6 bits. À titre d'exemple, à l'adresse 100110 (nombre décimal 38) correspond en sortie 00111000, la représentation DCB du nombre décimal 38.

Générateur de caractères Si vous avez pris la peine d'examiner attentivement les caractères alphanumériques (lettres, nombres, etc.) affichés sur un écran d'ordinateur, vous avez sans doute remarqué que chacun de ces caractères est formé d'un groupe de points. Selon le caractère à afficher, certains points brillent tandis que d'autres sont invisibles. Chaque caractère est constitué par un agencement particulier de points pris dans une matrice de points de 5 × 7 ou 7 × 9. L'agencement des points correspondant à chaque caractère peut être représenté par un code binaire (point brillant = 1, point invisible = 0).

Une *mémoire MORTE agissant comme générateur de caractères* garde en mémoire les codes des agencements des points correspondant à chaque caractère dans un emplacement adressé par le code ASCII de ce caractère. Par exemple, l'agencement des points pour la lettre « A » se trouve dans l'emplacement dont l'adresse est 1000001, où 1000001 est le code ASCII de A majuscule. Les mémoires MORTES génératrices de caractères sont utilisées partout où on doit afficher ou imprimer des caractères alphanumériques.

Générateur de fonctions Le générateur de fonctions est un circuit qui produit des formes d'ondes comme des ondes sinusoïdales, des ondes carrées, des ondes en dents de scie, des ondes en échelons. La figure 11-15 illustre comment avec une table de consultation en mémoire MORTE et un CNA on parvient à obtenir en sortie un signal sinusoïdal.

La mémoire MORTE conserve 256 valeurs de 8 bits, chacune correspondant à un point quelconque de la forme d'onde (une valeur de la tension sur l'onde sinusoïdale). Le compteur de 8 bits est sans cesse commandé par une horloge et parcourt ainsi séquentiellement toutes les adresses d'entrée de la mémoire MORTE. Au fur et à mesure que le compteur dénombre ses 256 états, les sorties de la mémoire MORTE fournissent les 256 points au CNA. La sortie de ce dernier est une forme d'onde qui passe par les 256 tensions analogiques correspondant aux 256 données conservées dans la mémoire. Le

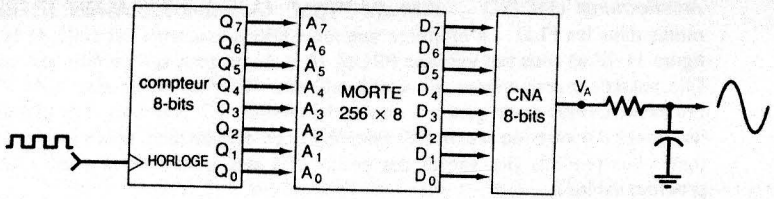


FIGURE 11-15 Générateur de fonctions constitué avec une mémoire MORTE et un CNA.

filtre passe-bas adoucit les transitions de la sortie du CNA afin de produire une forme d'onde lisse.

On retrouve des circuits semblables dans les générateurs de fonctions commerciaux. C'est d'ailleurs le même principe que l'on met en application dans certains synthétiseurs de la parole où les vibrations orales sont numérisées et conservées dans une mémoire MORTE.

QUESTIONS DE RÉVISION

1. Définissez microprogrammation.
2. Décrivez comment un ordinateur utilise un programme d'amorçage.
3. Qu'est-ce qu'un convertisseur de code?
4. Quel genre d'information est mémorisé dans un générateur de caractères (mémoire MORTE)?
5. Dites quels sont les principaux éléments d'un générateur de fonctions.

11-9 ÉLÉMENTS LOGIQUES PROGRAMMABLES

Les concepteurs logiciens disposent d'un large éventail de circuits intégrés dans lesquels sont matérialisés d'innombrables fonctions et montages logiques. En plus, ces puces sont fabriquées par de nombreux constructeurs et sont très bon marché. Pour toutes ces raisons, les concepteurs construisent tous les circuits et les systèmes qu'ils peuvent imaginer en câblant des circuits existants, et il n'y a rien qui permette de penser que cela va changer.

Parfois, certains problèmes surgissent quand on construit des circuits et des systèmes en recourant seulement qu'à des circuits existants. Certaines solutions doivent faire appel à des centaines, voire à des milliers de CI. Une aussi grande quantité de CI nécessite une plaquette imprimée de grande superficie, de même qu'un temps considérable pour l'insertion, le soudage et l'essai de toutes ces puces. En outre, le fabricant d'appareils doit conserver un imposant stock de tous les différents types de CI entrant dans les solutions retenues. Parvenir à réduire le nombre de CI dans une conception peut présenter nombre d'avantages: superficie de plaquette moindre, donc circuits imprimés plus petits et châssis moins encombrants; besoins en puissance plus faibles (et donc alimentations plus petites); procédés d'assemblage plus rapides et moins dispendieux; fiabilité accrue, puisqu'avec moins de CI, le risque de rupture d'une connexion est diminuée d'autant; enfin, dépannage plus aisé.

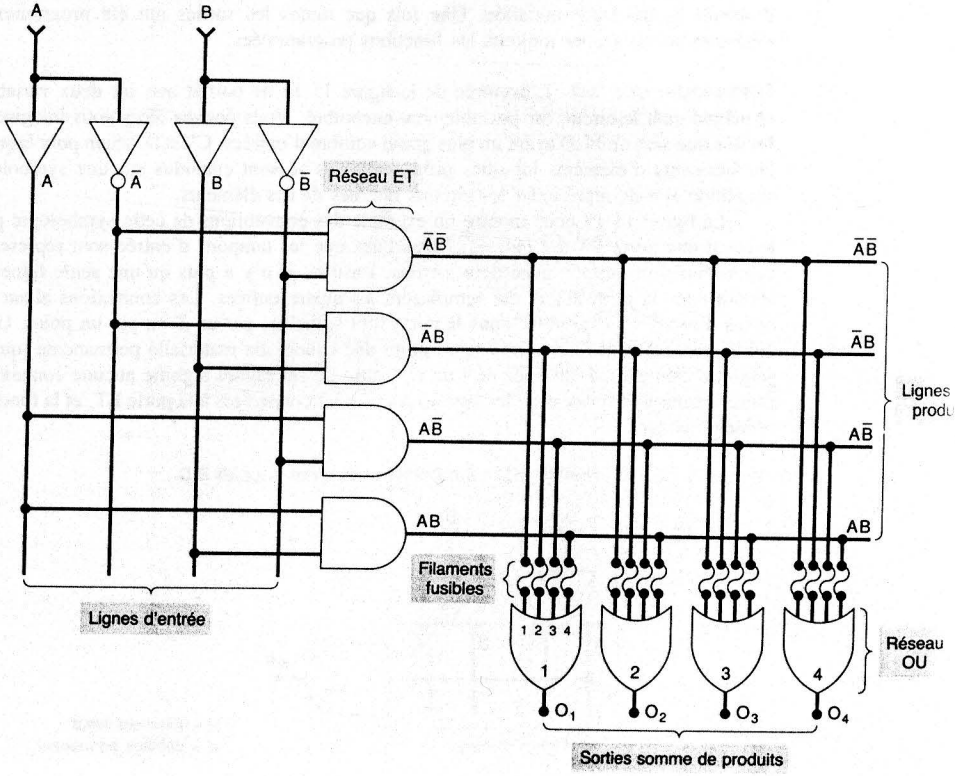
Pour parvenir à réduire le nombre de CI entrant dans une conception, il est nécessaire de mettre un nombre de plus en plus grand de fonctions dans une puce. Ceci, évidemment, a déjà été réalisé par les technologies LSI et VLSI pour des fonctions courantes comme les mémoires, les microprocesseurs, les synthétiseurs de la parole, les puces de calculatrice.

Ces dispositifs possèdent des centaines, même des milliers de portes toutes câblées en interne en vue de matérialiser un fonctionnement prévisible. Dans de nombreux projets d'étude, toutefois, il n'y a pas de solution LSI et VLSI toutes faites. C'est dans le cadre de tels projets que les concepteurs sont tenus d'assembler un grand nombre de puces SSI et MSI afin d'aboutir au système envisagé. Les récents développements apportés aux *éléments logiques programmables* (*PLD*) donnent aux concepteurs l'espoir de parvenir à remplacer plusieurs CI existants par un seul circuit intégré.

Un PLD est un CI qui renferme un très grand nombre de portes, de bascules et de registres qui sont raccordés dans une puce. Plusieurs de ces connexions sont des filaments fusibles qui peuvent être fondus. On dit qu'un tel CI est *programmable* parce que la fonction particulière que réalisera ce circuit est déterminée par la structure restante quand on aura fait fondre sélectivement certaines connexions et garder intactes les autres. Le processus de la « fonte du filament » peut être mené par le fabricant selon les spécifications du client, ou peut être effectué par le client même. Ce processus a reçu le nom de *programmation*, parce qu'en interconnectant les portes, les bascules et les registres, on obtient la maquette de circuit recherchée.

Principe de base La figure 11-16 illustre le principe de base mis en oeuvre dans tous les CI à logique programmable. On peut y voir un réseau de portes ET, et un réseau de

FIGURE 11-16 Exemple d'un élément logique programmable.



portes OU interconnectés pour pouvoir fournir quatre sorties, correspondant à n'importe quelle fonction logique des deux variables d'entrée A et B.

Chaque entrée alimente un tampon non inverseur et un tampon inverseur qui réalisent la forme directe et la forme complémentée de chaque variable. Ces sorties deviennent les *lignes d'entrée* du réseau de portes ET. Chaque porte ET est connectée à deux lignes d'entrée différentes, afin de générer ainsi un produit unique des variables d'entrée. Les sorties des portes ET sont appelées les *lignes des produits*.

Chacune des lignes des produits est raccordée à l'une des quatre entrées à filament fusible d'une porte OU. Quand tous les filaments restent intacts, la sortie de chacune des portes OU est 1. Voici la preuve de cette affirmation:

$$\begin{aligned} O_1 &= \overline{A}\overline{B} + \overline{A}B + A\overline{B} + AB \\ &= \overline{A}(\overline{B} + B) + A(\overline{B} + B) \\ &= \overline{A} + A = 1 \end{aligned}$$

Chacune des quatre sorties O_1, O_2, O_3 et O_4 peut être *programmée* afin d'obtenir n'importe quelle fonction de A et B, et cela en faisant fondre les filaments appropriés. Les PLD sont conçus pour qu'une entrée OU dont le filament a été fondu se comporte comme un 0 logique. Par exemple, si les filaments fusibles 1 et 4 de la porte OU 1 sont fondus, la sortie est alors:

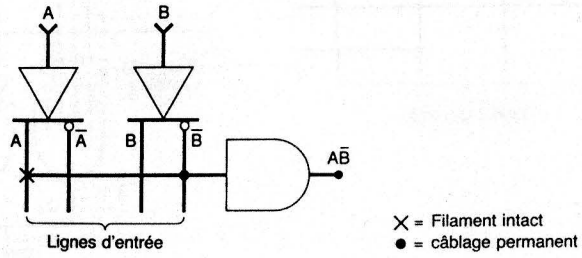
$$O_1 = 0 + \overline{A}B + A\overline{B} + 0 = \overline{A}B + A\overline{B}$$

Il est possible de programmer de la même manière chacune des entrées OU en vue d'obtenir la fonction souhaitée. Une fois que toutes les sorties ont été programmées, l'élément contient pour toujours les fonctions programmées.

Symbologie des PLD L'exemple de la figure 11-16 ne portait que sur deux variables, et malgré cela le circuit est passablement encombré. Vous pouvez alors vous imaginer le fouillis que sera un PLD ayant un plus grand nombre d'entrées. C'est la raison pour laquelle les fabricants d'éléments logiques programmables se sont entendus sur une symbologie simplifiée afin de représenter les circuits internes de ces éléments.

La figure 11-17 nous montre un exemple des conventions de cette symbologie pour le cas d'une porte ET à 4 entrées. Remarquez que les tampons d'entrée sont représentés comme un seul tampon avec deux sorties. Ensuite, il n'y a plus qu'une seule ligne qui aboutit dans la porte ET et qui remplacent les quatre entrées. Les connexions allant des lignes d'entrée des variables dans la porte sont signalées par un X ou par un point. Un X indique un filament fusible intact, un point une connexion matérielle permanente (qui ne peut être changée). L'absence de l'un et l'autre de ces signes signifie aucune connexion. Dans l'exemple de ci-dessus, les entrées A et $\overline{B}$ sont connectées à la porte ET, et la fonction générée est $A\overline{B}$.

FIGURE 11-17 Symbologie simplifiée pour les PLD.



*Architectures des PLD — **Élément PROM*** On retrouve plusieurs architectures courantes dans les PLD. La première que nous allons examiner est celle de la PROM. La figure 11-18 a) nous fait voir une PROM 16 × 4, qui peut agir comme une sorte de PLD. Elle possède quatre entrées qui sont totalement décodées par le réseau de ET; plus précisément, chaque porte génère l'un des 16 produits ET possibles. Les connexions allant des lignes d'entrées au réseau ET sont câblées en permanence, tandis que celles allant des lignes des produits (les sorties des portes ET) aux entrées de la porte OU sont toutes programmables.

La figure 11-18 b) nous montre comment il faudrait programmer une PROM pour obtenir quatre fonctions logiques spécifiées. Suivons les étapes de ce travail pour le cas $O_3 = AB + \overline{C}\overline{D}$. La première étape est de dresser la table de vérité illustrant le niveau de sortie souhaité pour O_3 relativement à toutes les combinaisons d'entrée possibles.

D	C	B	A	O_3
0	0	0	0	$1 \rightarrow \overline{D}\overline{C}\overline{B}\overline{A}$
0	0	0	1	$1 \rightarrow \overline{D}\overline{C}\overline{B}A$
0	0	1	0	$1 \rightarrow \overline{D}\overline{C}B\overline{A}$
0	0	1	1	$1 \rightarrow \overline{D}\overline{C}BA$
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	$1 \rightarrow \overline{D}CBA$
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	$1 \rightarrow D\overline{C}BA$
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	$1 \rightarrow DCBA$

Ensuite, écrivez sur un papier les produits ET pour lesquels les sorties doivent être 1. La sortie O_3 doit être la somme OU de ces produits. Seuls les filaments qui sont l'aboutissement des produits devant entrer dans la porte OU sont gardés intacts. Tous les autres sont fondus, comme le montre la figure 11-18 b). La même démarche est suivie pour établir l'état des filaments raccordés aux autres entrées de la porte OU.

La PROM peut servir à matérialiser n'importe quelle fonction des variables d'entrée, puisqu'elle permet d'obtenir n'importe lequel des produits ET possibles. En général, toutes les fonctions qui utilisent chacune des combinaisons d'entrée possibles conviennent bien à une architecture PROM. Cependant, les PROM cessent d'être une solution réaliste quand il y a beaucoup de variables d'entrée, puisque le nombre de filaments fusibles double à chaque nouvelle variable d'entrée.

Une PROM réelle qui sert souvent de PLD est le AM27S13, ayant une capacité de 512 × 4, fabriquée en technologie TTL Schottky rapide. Étant donné que $512 = 2^9$, cette PROM possède neuf entrées d'adresses et quatre entrées de donnée. On peut donc programmer le AM27S13 afin de générer quatre sorties, chacune étant une fonction logique des neuf entrées différentes.

Élément PAL L'architecture d'un élément PROM convient bien aux applications qui, dans les fonctions mises en oeuvre, requièrent toutes les combinaisons possibles des entrées.

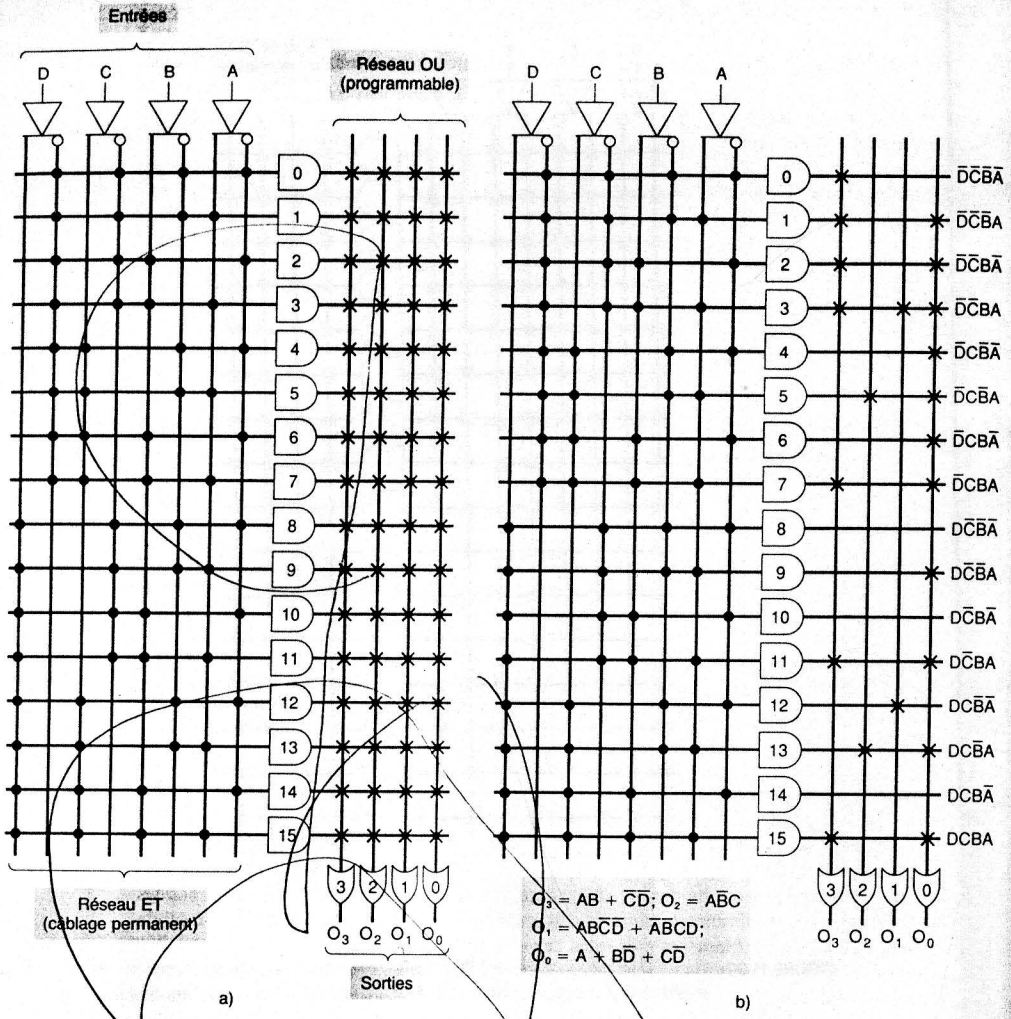


FIGURE 11-18 a) L'architecture d'une PROM lui permet d'être un élément PLD; b) des filaments fusibles sont fondus afin de programmer les sorties correspondant à des fonctions précises.

C'est le cas notamment des convertisseurs de codes et des tables de mémorisation de données. À l'opposé, on trouve une foule d'applications qui n'ont pas besoin de programmer la totalité des combinaisons des entrées. Les fonctions logiques illustrées à la figure 11-18 b), par exemple, n'utilisent pas toutes les lignes des produits de la matrice ET. Pour ce genre d'applications, on a développé une classe d'élément PLD appelé les éléments PAL (*programmable array logic*). L'architecture d'un élément PAL se démarque légèrement de celle d'un élément PROM, comme l'illustre la figure 11-9 a).

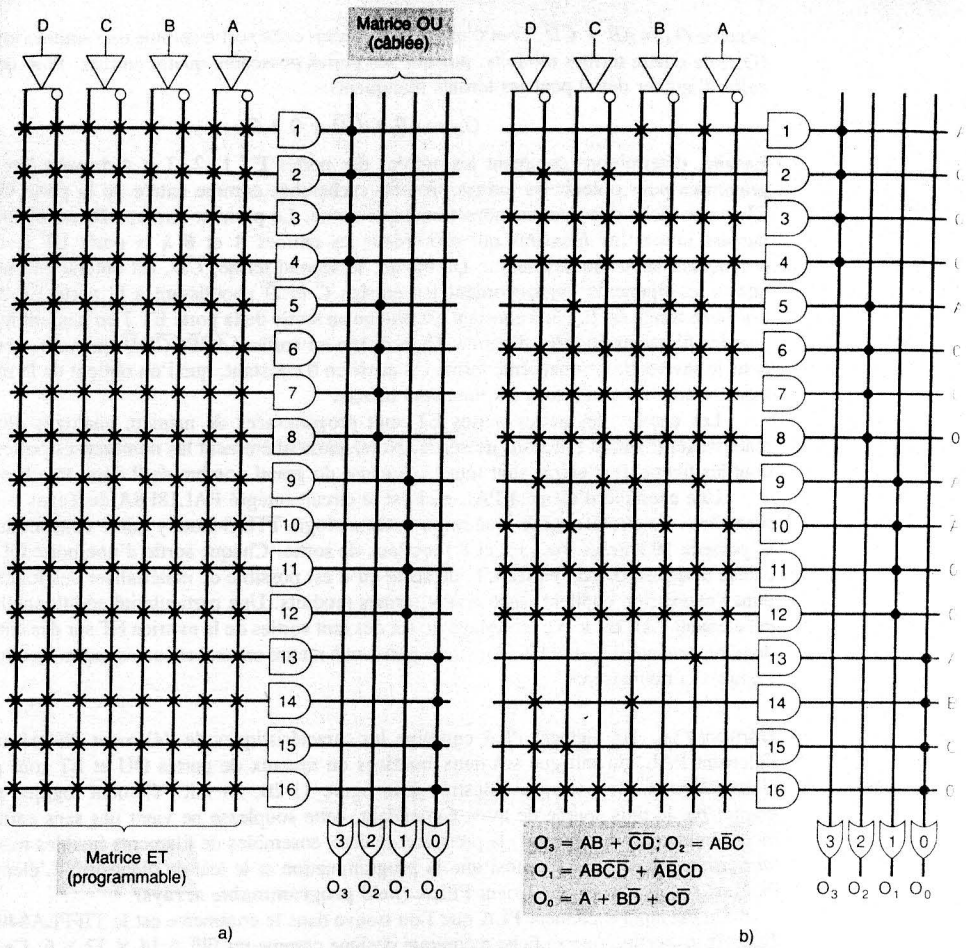


FIGURE 11-19 a) Architecture type d'un élément PAL; b) le même PAL programmé pour matérialiser certaines fonctions.

L'élément PAL possède les mêmes matrices de ET et de OU que l'élément PROM, mais dans le premier seules les portes ET sont programmables, tandis que les portes OU sont câblées. Chaque porte ET peut donc être programmée pour produire n'importe quel produit que l'on veut des quatre variables et de leur forme complémentée. Chaque porte OU a comme entrées des conducteurs permanents provenant des sorties de quatre portes ET. Cela a pour effet de limiter la fonction de sortie à quatre termes produits. Si l'expression d'une fonction renferme plus de quatre termes produits, on ne peut utiliser cet élément PAL pour la matérialiser; il faudra recourir à un élément ayant plus de quatre entrées. Par contre, si l'expression compte moins de quatre termes produits, on peut mettre les sorties inutilisées à 0.

La figure 11-19 b) nous fait voir comment cet élément PAL est programmé pour fournir en sortie quatre fonctions logiques spécifiées. Examinons la façon de procéder pour

la sortie $O_3 = AB + \overline{C}\overline{D}$. Tout d'abord, exprimons cette sortie comme une somme logique (OU) de quatre termes produits, puisque les portes possèdent quatre entrées. Pour cela, il suffit d'ajouter des 0 pour les termes manquants:

$$O_3 = AB + \overline{C}\overline{D} + 0 + 0$$

Ensuite, déterminons comment les entrées des portes ET 1, 2, 3 et 4 doivent être programmées pour obtenir les termes produits recherchés comme entrée de la porte OU 3. C'est une tâche que nous menons terme par terme. Le premier terme, AB, est obtenu en laissant intacts les filaments qui raccordent les entrées A et B à la porte ET 1, et en provoquant la fusion des autres. De même, le second terme, $\overline{C}\overline{D}$, est obtenu en laissant intacts les filaments qui raccordent les entrées $\overline{C}$ et $\overline{D}$ aboutissant à la porte ET 2. Le troisième terme est 0. Un 0 constant est obtenu en sortie de la porte ET 3 en laissant intacts tous les filaments fusibles d'entrée. Cela donne en sortie $A\overline{A}B\overline{B}C\overline{C}D\overline{D}$ qui vaut, comme nous le savons 0. Le quatrième terme est aussi un 0 constant, que l'on obtient de la même manière en conservant tous les filaments intacts.

Les entrées des autres portes ET sont programmées de manière analogue afin de matérialiser d'autres fonctions de sortie. Notez particulièrement les nombreuses portes ET dont les filaments d'entrée sont intacts, à cause du grand nombre de 0 dont on a besoin.

Une exemple d'élément PAL réel est le circuit intégré PAL18L8A de Texas Instruments Inc. Ce circuit est fabriqué selon la technologie TTL Schottky faible consommation et possède 10 entrées logiques et 8 fonctions de sortie. Chaque sortie d'une porte OU est câblée à sept sorties de portes ET, de sorte qu'il est possible de matérialiser des fonctions dont l'expression contient jusqu'à sept termes produits. Une particularité additionnelle de cet élément PAL est le rétrocouplage de six des huit sorties de la matrice ET sur des entrées de n'importe quelle porte ET. Ainsi, on parvient à mettre en oeuvre toutes sortes de circuits logiques combinatoires.

Élément PLA Un élément PLA combine les caractéristiques de l'élément PROM et de l'élément PAL, du fait que ses deux matrices ou réseaux de portes OU et ET sont programmables. Cette propriété, illustrée à la figure 11-20, en fait l'élément logique programmable le plus souple de tous. Cependant, cette souplesse ne vient pas sans certains inconvénients. Premièrement, la présence de deux ensembles de filaments fusibles rend la fabrication plus complexe, ainsi que la programmation et le test du dispositif. L'élément PLA est souvent désigné élément FPLA (**field programmable arrays**).

Un exemple d'élément PLA que l'on trouve dans le commerce est le TIFPLA840 de Texas Instruments, que sa fiche technique désigne comme un FPLA $14 \times 32 \times 6$. Ce qui signifie 14 variables d'entrée, 32 portes ET pour réaliser les termes produits de ces variables et 6 portes OU pour faire la somme de n'importe quelle combinaison des 32 sorties ET. Ce CI est en outre doté d'une fonction appelée *polarité de la sortie programmable*, qui donne au logicien la possibilité d'inverser n'importe quelle sortie. Une description graphique de cette fonction est donnée à la figure 11-21, dans laquelle une porte OU exclusif et un filament fusible de polarité complètent la sortie de la porte OU. La porte OU exclusif laisse passer la sortie quittant la porte OU O_3 sans modification si le filament fusible est intact (logique 0). Par contre, si le filament est fondu (logique 1), la porte OU exclusif inverse la sortie de la porte OU associée. Cette polarisation des sorties est une caractéristique disponible dans bien des éléments PAL et FPLA.

Autres fonctions des éléments PLD Voici un certain nombre de fonctions qui viennent compléter la majeure partie des éléments logiques programmables: bascules, mémoires, registres d'entrée et registres de sortie. Très souvent, les données de fonctionnement de ces éléments sont programmables, tout comme le raccordement à d'autres éléments dans

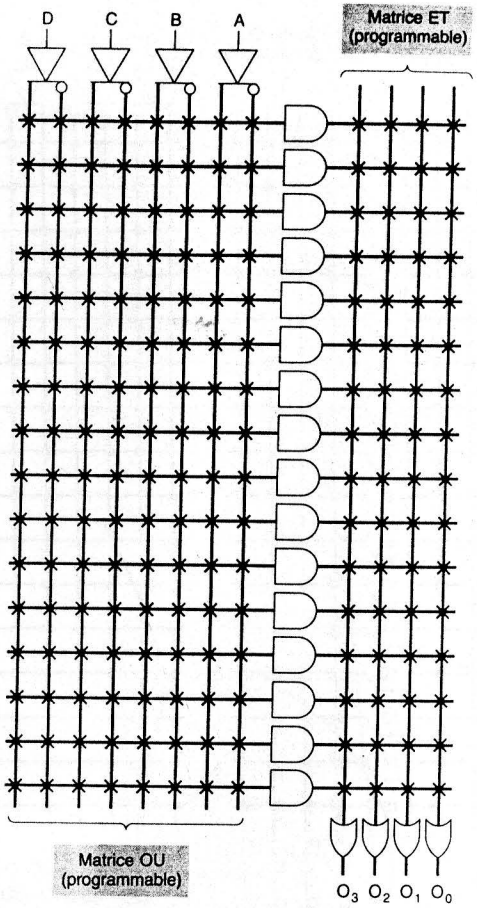


FIGURE 11-20 Architecture d'un élément PLA illustrant la matrice ET programmable et la matrice OU programmable; c'est l'élément PLD le plus souple de tous.

la puce. Le concepteur logicien a donc tout le loisir de créer sans contrainte des compteurs ou d'autres circuits logiques séquentiels.

Programmation Avec les premiers éléments PLD, le logicien devait produire une carte des *fusibles* sur laquelle étaient indiqués les filaments à faire fondre et ceux à conserver; il envoyait cette carte au fabricant de l'élément PROM, PAL ou FPLA. Celui-ci programmait ensuite le dispositif conformément à la carte, testait le circuit et retournait le tout au logicien. Depuis quelques années, l'arrivée de programmateurs relativement peu coûteux incite beaucoup d'utilisateurs à programmer eux-mêmes leurs PLD. On retrouve sur le marché des programmateurs universels pouvant programmer aussi bien des PROM, des PAL que des FPLA. Le circuit à programmer est enfiché dans le support du programmateur,

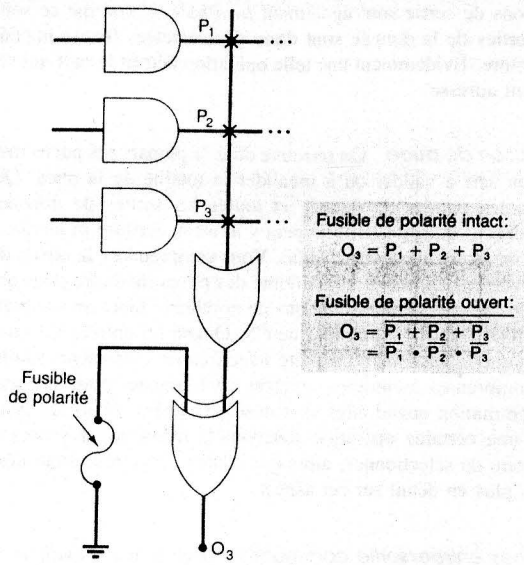


FIGURE 11-21 Nombre d'éléments PLD dotés de la fonction de polarisation des sorties donnent l'avantage de pouvoir intervertir n'importe quelle fonction de sortie.

et ce dernier ouvre les connexions d'après les instructions de l'utilisateur, puis il teste le circuit définitif.

Les données de programmation et de mise à l'essai sont généralement obtenues au moyen d'un logiciel fonctionnant sur un micro-ordinateur PC ordinaire. Au moyen de ce logiciel, l'utilisateur entre les données qui décrivent les fonctions logiques que l'on veut matérialiser dans un circuit PLD, de même que celles qui indiquent quels tests faire subir au circuit. Le logiciel génère ensuite une carte des fusibles et les données des tests, puis il transmet, par l'entremise d'un câble, toutes ces informations dans la mémoire du programmateur. Muni des données nécessaires, le programmateur se charge de la programmation et des tests. Quand tout est terminé, le programmateur indique si le circuit a réussi ou échoué la procédure de test. Si tout est concluant, le circuit est retiré du support et placé dans le circuit prototype pour des essais plus poussés.

PLD effaçables Les PLD que l'on vient de décrire sont tous programmés par la fusion de filaments fusibles. Une fois que les filaments sont ouverts, il n'y a plus moyen de reconstruire la connexion. C'est pourquoi une erreur de programmation ou un changement de conception nous force à tout recommencer à neuf. Plusieurs constructeurs ont tenté de corriger cet inconvénient en développant des PLD effaçables et reprogrammables indéfiniment. On les appelle *éléments logiques effaçables programmables* (*EPLD*), et le travail de programmation et d'effacement est très semblable à ce qui se fait dans les EPROM et les EEPROM.

QUESTIONS DE RÉVISION

1. Qu'est-ce qu'un élément logique programmable (PLD)?
2. Quelle est la sortie de O_1 du dispositif de la figure 11-16 si les fusibles 1 et 2 ont été fondus?
3. Que signifie un *X* dans un schéma de PLD?
4. Que signifie un point dans un schéma de PLD?
5. Vérifiez que pour les fonctions O_2, O_1 et O_0 de la figure 11-18 b), les bons filaments ont été fondus.
6. Quelle est la différence entre l'architecture PAL et l'architecture PROM?
7. Vérifiez que pour les fonctions O_2, O_1 et O_0 de la figure 11-19 b), les bons filaments ont été fondus.
8. Qu'est-ce qui différencie l'architecture d'un élément FPLA de celles d'une PROM et d'un PAL?
9. Indiquez les fonctions que réalise un programmateur de PLD.

11-10 MÉMOIRES VIVES À SEMICONDUCTEURS

Rappelons qu'une mémoire VIVE est une mémoire dans laquelle on peut écrire ou lire des données, et c'est la raison pour laquelle elle est différente d'une mémoire MORTE.

Les mémoires VIVES que l'on retrouve dans les ordinateurs servent au stockage *temporaire* des programmes et des données. Le contenu de certains emplacements adressables de mémoire VIVE change continuellement à mesure que progresse l'exécution du programme. Les mémoires VIVES doivent donc avoir des temps de cycle pour la lecture et l'écriture très courts, afin de ne pas ralentir le fonctionnement de l'ordinateur.

Le principal inconvénient des mémoires VIVES est la volatilité, c'est-à-dire que les données qu'elles contiennent sont perdues si la tension est interrompue ou enlevée. Toutefois, certaines mémoires VIVES consomment tellement peu de courant en mode de repos (lorsqu'elles ne font pas l'objet d'une opération d'écriture ou de lecture), qu'elles peuvent être alimentées à partir d'une pile lorsque l'alimentation principale est interrompue. Le principal avantage des mémoires VIVES est, comme on s'en doute bien, leur faculté d'être écrites ou lues rapidement et avec une égale facilité.

L'analyse suivante des mémoires VIVES repose en partie sur de la matière déjà couverte dans notre étude des mémoires MORTES, puisque ces deux types de mémoire partagent plusieurs principes de base.

11-11 ARCHITECTURE DES MÉMOIRES VIVES

Il peut être utile de considérer une mémoire VIVE comme un assemblage de registres, comme on l'a fait pour la mémoire MORTE, chaque registre emmagasinant une donnée et ayant une adresse exclusive. Typiquement, les mémoires VIVES ont une capacité de 1K, 4K, 8K, 16K ou 64K, 128K, 256K et 1024K et des longueurs de mots de 1, 4 ou 8 bits. Comme nous le verrons plus loin, la capacité et la longueur peuvent être étendues en associant des puces mémoires.

La figure 11-22 nous montre une architecture simplifiée d'une mémoire VIVE qui peut conserver 64 mots de 4 bits chacun (c'est donc une mémoire 64 × 4). À ces mots correspondant des adresses allant de 0 à 63_{10}. Pour choisir un des 64 emplacements adressables en vue d'effectuer une lecture ou une écriture, on doit appliquer à un circuit décodeur un code d'adresse binaire. Étant donné que $64 = 2^6$, il nous faut un décodeur

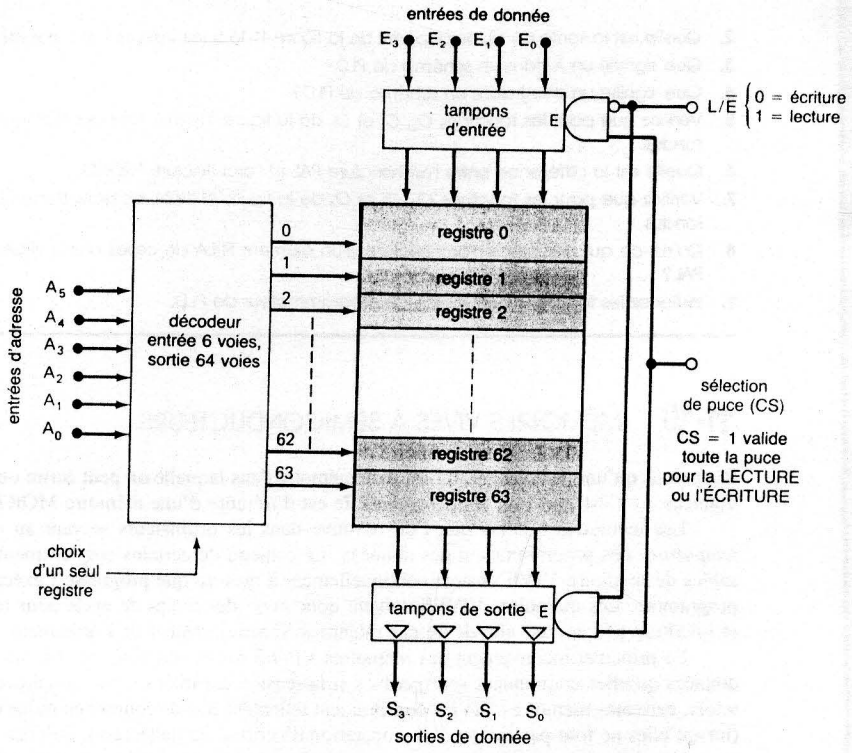


FIGURE 11-22 **Test des mémoires MORTES**

dont l'entrée à 6 bits. Chaque adresse rend active une sortie du décodeur, qui à son tour valide le registre correspondant. Par exemple, supposons que l'adresse appliquée soit:

$$A_5A_4A_3A_2A_1A_0 = 011010$$

Puisque $011010_2 = 26_{10}$, la sortie 26 du décodeur passe au niveau HAUT, ce qui a pour effet de choisir le registre 26 pour une opération de lecture ou d'écriture.

Opération de lecture Le code d'adresse repère un registre de la puce mémoire en vue d'effectuer une lecture ou une écriture. Pour *lire* le contenu du registre choisi, l'entrée LECTURE/ÉCRITURE ($L/\overline{E}$) doit être à 1. En outre, l'entrée SÉLECTION DE PUCE (*CS*) doit également être à son niveau vrai (dans ce cas-ci à 1). La combinaison $L/\overline{E} = 1$ et $CS = 1$ valide les tampons de sortie, ce qui place le contenu du registre choisi sur les 4 sorties de donnée. $L/\overline{E} = 1$ *invalide* également les tampons d'entrée de sorte qu'aucune entrée de donnée ne vient affecter la mémoire durant une opération de lecture.

Opération d'écriture Pour écrire un mot de 4 bits dans le registre choisi, il faut que $L/\overline{E} = 0$ et $CS = 1$. Cette combinaison *valide* les tampons d'entrée et le nouveau mot de

4 bits appliqué aux entrées de donnée est chargé dans le registre. Quand $L/\overline{E} = 0$, les tampons de sortie sont également *invalidés*, et comme ce sont des dispositifs trois états, les sorties de la donnée sont donc déconnectées (haute impédance) durant une opération d'écriture. Évidemment une telle opération détruit le mot qui se trouvait déjà dans l'emplacement adressé.

Sélection de puce On retrouve dans la plupart des puces mémoires au moins une entrée *CS* qui sert à valider ou à invalider la totalité de la puce. Quand la puce est invalidée, toutes les entrées de donnée et toutes les sorties de donnée sont déconnectées (haute impédance) de sorte qu'il ne peut y avoir ni écriture ni lecture. Dans ce mode, le contenu de la mémoire n'est pas modifié. Vous comprendrez la raison d'être de l'entrée *CS* lorsque nous verrons comment on combine des puces mémoire pour obtenir des mémoires de plus grande capacité. Prenez note que de nombreux fabricants appellent cette entrée *CE* (VALIDATION DE PUCE) plutôt que *CS*. Quand les entrées *CS* ou *CE* sont à leur niveau vrai, on dit que la puce mémoire a été *sélectionnée*; autrement, on dit qu'elle est *désélectionnée*. De nombreuses mémoires intégrées sont conçues pour se mettre dans un mode de faible consommation quand elles sont désélectionnées. Dans les systèmes de mémoire étendus, pour une certaine opération touchant la mémoire, il y aura une ou plusieurs puces de mémoire de sélectionner, alors que toutes les autres seront désélectionnées. Nous reviendrons plus en détail sur cet aspect.

Broches entrée/sortie communes Dans le but d'avoir le moins de broches possibles sur un boîtier de CI, les fabricants combinent souvent les fonctions d'entrée et de sortie de donnée sur les mêmes broches d'entrée/sortie (E/S). L'entrée $L/\overline{E}$ commande la fonction des broches E/S. Durant une opération de lecture, les broches E/S agissent comme des sorties de donnée afin de reproduire le contenu d'un emplacement adressé. Durant une opération d'écriture, les broches E/S agissent comme des entrées de donnée.

La figure 11-22 nous montre pourquoi on choisit une telle solution. Si on avait des broches d'entrée et de sortie distinctes, il faudrait au total 18 broches à ce boîtier (en comptant la masse et l'alimentation). En utilisant des broches E/S, il n'en faut plus que 14. Cette économie de broches est encore plus importante lorsque les puces ont un mot beaucoup plus long.

EXEMPLE 11-9

Le boîtier 2147 est une mémoire VIVE NMOS de 4K × 1 dotée d'entrées et de sorties distinctes et d'une seule entrée de validation de puce vraie au niveau BAS. Dessinez le symbole logique de cette puce en indiquant la fonction de chaque broche.

Solution Le symbole logique recherché est illustré à la figure 11-23 a).

EXEMPLE 11-10

Le boîtier 2168 est une mémoire VIVE NMOS de 4K × 4 ayant des broches entrée/sortie communes et une entrée de validation de puce vraie au niveau BAS. Dessinez son symbole logique.

Solution Le symbole logique recherché est illustré à la figure 11-23 b).

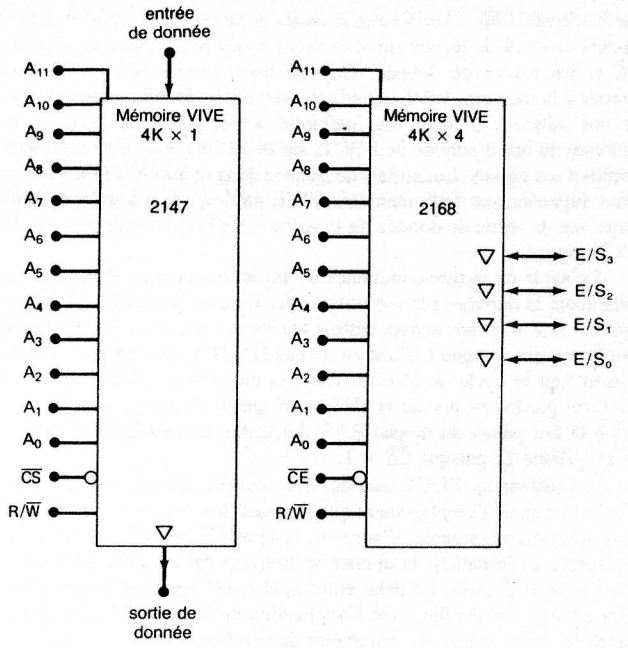


FIGURE 11-23 a) Symbole logique d'une mémoire VIVE 2147; b) d'une mémoire VIVE 2168.

QUESTIONS DE RÉVISION

1. Décrivez les conditions d'entrée qui commandent la lecture d'un mot placé dans un emplacement particulier de la mémoire VIVE.
2. Dites pourquoi certaines puces de mémoire VIVE ont des broches entrée/sortie communes.
3. Indiquez le nombre de broches qu'il faut pour une mémoire VIVE 256 × 8 ayant des broches E/S communes et une entrée *CS*.

11-12 MÉMOIRES VIVES STATIQUES (SRAM)

Le fonctionnement des mémoires VIVES que l'on a décrit jusqu'à ce point-ci concernait des mémoires VIVES *statiques*, c'est-à-dire des mémoires qui conservent les données tant que la puce est sous tension. Les cellules des mémoires VIVES statiques sont essentiellement des bascules qui demeurent dans un état donné (stockage d'un bit) indéfiniment tant que le circuit demeure alimenté. Plus tard nous décrirons les mémoires VIVES *dynamiques* qui mémorisent les données sous forme de charges dans des condensateurs. Dans ce dernier cas, les données se détériorent graduellement à mesure que se décharge le condensateur, et il est alors nécessaire de rafraîchir périodiquement les données (c'est-à-dire de recharger le condensateur).

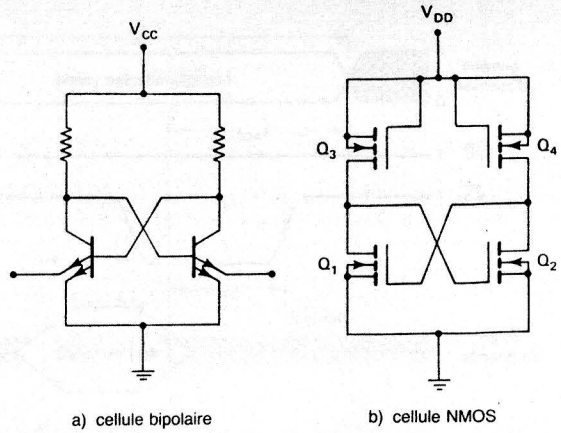
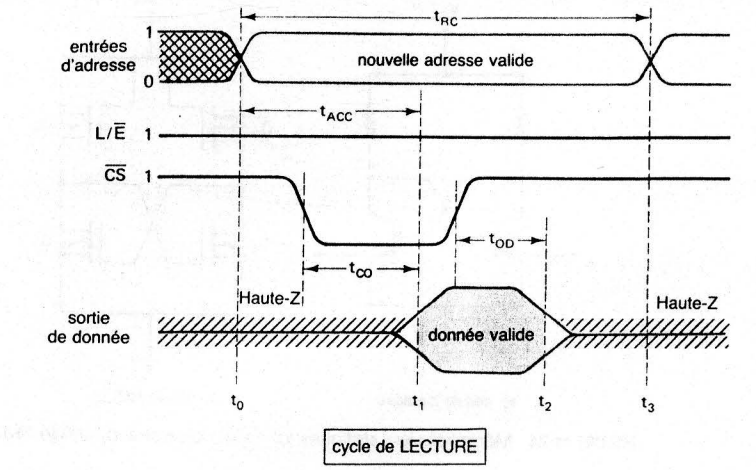


FIGURE 11-24 Mémoires vives statiques: a) cellule bipolaire, b) cellule NMOS.

On trouve sur le marché des mémoires VIVES statiques aussi bien bipolaires que MOS, quoique la majeure partie des mémoires VIVES utilisées soit de type NMOS ou CMOS. Comme on l'a dit précédemment, les mémoires bipolaires sont plus rapides (même si les mémoires NMOS se rapprochent continuellement), tandis que les dispositifs MOS permettent d'obtenir de plus grandes capacités. La figure 11-24 compare une cellule mémoire statique bipolaire typique et une cellule mémoire statique NMOS. La cellule bipolaire contient deux transistors bipolaires ainsi que deux résistances, alors que la cellule NMOS renferme 4 transistors MOS à canal-*N*. La cellule bipolaire s'étend sur une plus grande superficie de la puce que la cellule MOS parce qu'un transistor bipolaire est plus complexe qu'un transistor MOS et parce que la cellule bipolaire a besoin de résistances distinctes, alors que la cellule MOS utilise des transistors MOS comme résistances (Q_3 et Q_4). Une cellule mémoire CMOS est analogue à une cellule NMOS, sauf les transistors Q_3 et Q_4 qui sont des transistors MOS à canal-*P*. La mémoire qui en résulte consomme moins d'énergie, mais au détriment d'une puce plus complexe, de sorte que les capacités réalisables avec la technologie CMOS se situent à peu près à mi-chemin entre la technologie bipolaire et la technologie NMOS.

Synchronisation des mémoires VIVES statiques Les circuits intégrés de mémoire VIVE servent le plus souvent de mémoire interne dans un ordinateur. L'unité centrale de traitement (UCT) effectue sans arrêt des opérations de lecture et d'écriture sur cette mémoire à un rythme très rapide, limité seulement par les possibilités de cette unité. Les puces mémoires qui interagissent avec l'UCT doivent être suffisamment rapides pour répondre aux commandes de lecture et d'écriture de l'UCT; c'est pourquoi le concepteur d'ordinateur doit prendre en considération diverses caractéristiques de synchronisation des mémoires VIVES.

Les mémoires VIVES n'ont pas toutes les mêmes caractéristiques de synchronisation, quoique la plupart soient semblables. C'est la raison pour laquelle nous allons utiliser un ensemble de caractéristiques typiques afin d'illustrer nos propos. La désignation des différents paramètres de synchronisation peut varier d'un fabricant à l'autre, mais le sens de chaque paramètre se déduit facilement des chronogrammes de mémoire que l'on trouve dans les fiches techniques des mémoires VIVES. La figure 11-25 nous montre les chronogrammes pour un cycle de lecture complet et un cycle d'écriture complet d'une puce de mémoire VIVE typique.



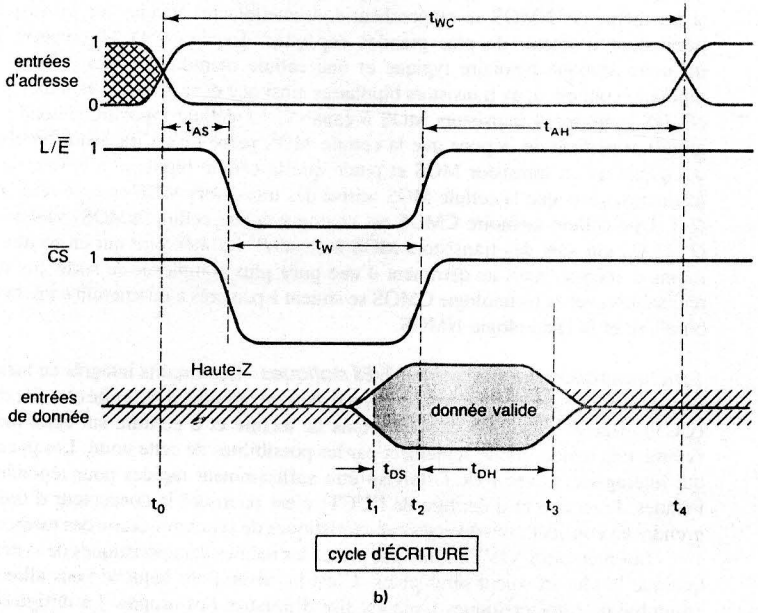


FIGURE 11-25 Chronogramme typique d'une mémoire VIVE statique: a) cycle de LECTURE; b) cycle d'ÉCRITURE.

Cycle de LECTURE Les formes d'ondes de la figure 11-25 a) montrent ce qui se produit pendant un cycle de lecture sur les entrées d'adresse, l'entrée de sélection de puce, l'entrée $L/\overline{E}$ et les sorties de donnée. Comme nous l'avons dit, l'UCT transmet ces signaux d'entrée à la mémoire VIVE quand elle veut lire la donnée contenue dans une case spécifiée par une adresse. Bien qu'une mémoire VIVE puisse recevoir de nombreuses entrées d'adresse du bus d'adresse de l'UCT, sur le schéma nous n'en représentons que deux pour simplifier les choses. Les sorties de donnée de la mémoire VIVE sont aussi illustrées; nous allons supposer que cette mémoire VIVE particulière n'a qu'une sortie de donnée. Rappelons que la sortie de donnée de la mémoire VIVE est branchée sur le bus de donnée de l'UCT (figure 11-5).

Le cycle de lecture commence à l'instant t_0. Avant cet instant, les entrées d'adresse contiennent la dernière adresse utilisée par l'opération précédente. Comme la sélection de la puce VIVE n'est pas active, cette « ancienne » adresse n'est pas décodée par cette puce. Remarquez que la ligne $L/\overline{E}$ est au niveau HAUT avant l'instant t_0 et demeure à ce niveau pendant tout le cycle de lecture. Dans la majorité des mémoires, la ligne $L/\overline{E}$ est normalement gardée au niveau HAUT, sauf quand on veut procéder à une écriture, auquel cas on la fait passer au niveau BAS. La sortie de donnée de la mémoire VIVE est dans son état Haute-Z, puisque $\overline{CS} = 1$.

À l'instant t_0, l'UCT impose une nouvelle adresse sur les entrées de la mémoire VIVE; il s'agit de l'emplacement que l'on veut lire. Après un délai suffisant pour permettre la stabilisation des signaux d'adresse, la ligne $\overline{CS}$ est activée. La mémoire VIVE répond en plaçant, à l'instant t_1, la donnée se trouvant dans la case adressée sur les conducteurs de sortie de la donnée. Le délai entre t_0 et t_1 est appelé le temps d'accès, t_{ACC}, et il est défini comme l'intervalle entre l'application de la nouvelle adresse et l'apparition d'une donnée de sortie valide. Le paramètre de synchronisation, t_{CO}, est le temps que prend la sortie de la mémoire VIVE, une fois que $\overline{CS}$ est activé, pour passer de son état Haute-Z à celui où elle contient une donnée valide.

À l'instant t_2, le signal $\overline{CS}$ revient au niveau HAUT, et les sorties de la mémoire VIVE repassent dans leur état Haute-Z après un intervalle t_{CO}. Ainsi, la donnée mémorisée se trouve sur le bus de donnée entre les instants t_1 et t_3. L'UCT peut la récupérer sur le bus à tout moment durant cet intervalle. Dans nombre d'ordinateurs, l'UCT exploite le FM du signal $\overline{CS}$ à l'instant t_2 pour verrouiller la donnée dans un de ses registres.

La durée du cycle de lecture complet, mesuré par t_{RC}, s'étend de t_0 à t_4, moment où l'UCT change la valeur de l'adresse afin d'amorcer un nouveau cycle de lecture ou d'écriture.

Cycle d'ÉCRITURE La figure 11-25 b) illustre la séquence de signaux qui intervient lors d'un cycle d'écriture, à partir du moment où l'UCT fournit la nouvelle adresse à la mémoire VIVE. L'unité centrale attaque les lignes $L/\overline{E}$ et $\overline{CS}$ et leur impose un niveau BAS, après un délai t_{AS}, appelé temps d'établissement de l'adresse. Ce délai laisse le temps aux décodeurs d'adresse de la mémoire VIVE de réagir à la nouvelle adresse. Les conducteurs $L/\overline{E}$ et $\overline{CS}$ restent au niveau BAS pendant un intervalle t_W, appelé temps d'écriture.

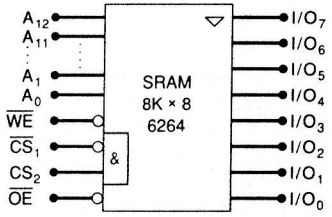
Durant le temps d'écriture, l'UCT en t_1 applique une donnée valide sur le bus de donnée qui sera transférée dans la case adressée. Cette donnée doit rester sur les entrées de la mémoire VIVE au moins la durée t_{DS} avant la désactivation de $L/\overline{E}$ et $\overline{CS}$ en t_2, et au moins la durée t_{DH} après cette désactivation. La durée t_{DS} est le temps d'établissement de la donnée, et la durée t_{DH} le temps de maintien de la donnée. De la même façon, les entrées d'adresse doivent rester stables pendant une durée d'établissement, t_{AH}, après t_2. Si toutes ces conditions d'établissement et de maintien ne sont pas respectées, il est impossible d'être certain que l'opération d'écriture s'est bien déroulée.

La durée du cycle d'écriture complet, mesurée par t_{WC}, s'étend de t_0 à t_4, moment où l'UCT change la valeur de l'adresse afin d'amorcer un nouveau cycle de lecture ou d'écriture.

La durée du cycle de lecture, t_{RC}, et la durée du cycle de lecture, t_{WC}, sont les paramètres qui conditionnent directement la vitesse de fonctionnement de la mémoire. Par exemple, dans une application réelle, une UCT lira souvent des mots en mémoire successifs les uns après les autres. Si la durée de lecture de la mémoire est 50 ns, l'UCT lit donc un mot en 50 ns, c'est-à-dire qu'elle pourra lire 20 millions de mots par seconde.

Puce SRAM réelle Une puce SRAM que l'on trouve dans le commerce est la CMOS 6264, plus précisément une 8K × 8 dont les durées des cycles de lecture et d'écriture sont 100 ns et la consommation dans l'état de veille n'est que 0,1 mW. Le symbole logique de la 6264 est montré à la figure 11-26. Remarquez les 13 entrées d'adresse, ce qui autorise l'accès à $2^{13} = 8192 = 8K$, et les huit lignes I/O. Les quatre conducteurs de commande fixent le mode opératoire du dispositif, conformément à la table associée.

L'entrée $\overline{WE}$ est l'équivalent de l'entrée $L/\overline{E}$ utilisée dans la description précédente. Un niveau BAS sur cette borne écrit la donnée dans la mémoire si cette puce a été sélectionnée — pour cela il faut que les deux entrées de sélection de puce soient actives. Notez le recours au signe & pour signaler que les *deux* entrées doivent être actives. Un niveau HAUT sur $\overline{WE}$ commande l'opération d'écriture à la condition que cette puce mémoire soit sélectionnée et que les tampons de sortie soient validés par $\overline{OE}$ = BAS. Quand elle n'est pas sélectionnée, cette puce est dans son mode de veille (faible consommation), et aucune entrée n'agit sur elle.



	Entrées				
Mode	$\overline{WE}$	$\overline{CS_1}$	CS_2	$\overline{OE}$	Broches I/O
LECTURE	1	0	1	0	$DONNÉE_{SORTIE}$
ÉCRITURE	0	0	1	X	$DONNÉE_{ENTRÉE}$
Sortie invalidée	1	X	X	1	Haute-Z
Non sélectionnée	X	1	X	X	Haute-Z
(pas de consommation)	X	X	0	X	

X = indifférent

FIGURE 11-26 Symbole de la SRAM CMOS 6264 et sa table de définition du mode.

QUESTIONS DE RÉVISION

1. Quelles sont les différences entre une cellule VIVE statique et une cellule VIVE dynamique?
2. Quelle technologie de fabrication des mémoires produit les éléments qui consomment le moins d'énergie?
3. Quels dispositifs placent la donnée sur le bus de donnée durant le cycle de lecture?
4. Quels dispositifs placent la donnée sur le bus de donnée durant le cycle d'écriture?
5. Quels paramètres de synchronisation de la mémoire VIVE conditionnent la vitesse de travail?
6. *Vrai ou faux:* Un niveau BAS sur $\overline{OE}$ active les tampons de sortie d'une SRAM 6264 à la condition que les deux entrées de sélection de puce soient validées.

11-13 MÉMOIRE VIVE DYNAMIQUE (DRAM)

Les mémoires VIVES dynamiques sont fabriquées au moyen de la technologie MOS et elles se distinguent par leur grande capacité, leur faible consommation et leur vitesse moyenne. Comme nous l'avons signalé plus tôt et contrairement aux mémoires VIVES statiques qui conservent leurs données dans des bascules, ces mémoires stockent les 1 et les 0 sous forme de charges dans de minuscules condensateurs MOS (généralement de l'ordre de quelques picofarads). À cause de l'épuisement de ces charges en fonction du temps, il est indispensable de recharger périodiquement les cellules mémoires des VIVES dynamiques; on parle alors de **rafraîchissement** des DRAM. De façon caractéristique, on doit les rafraîchir à toutes les 2 à 10 ms, sans quoi la donnée est perdue.

Cette nécessité de rafraîchir les condensateurs désavantage les mémoires VIVES dynamiques, puisque cela impose de nouvelles contraintes à la conception du système de mémoire. Jusqu'à tout dernièrement, les concepteurs de système étaient obligés d'ajouter des circuits supplémentaires pour réaliser le rafraîchissement à intervalles réguliers entre les périodes de lecture ou d'écriture. À présent, il existe deux solutions qui éliminent en partie cet inconvénient des DRAM. D'abord, il y a la *i*DRAM (VIVE dynamique intégrée) pour les mémoires relativement petites (< 64 kmots), qui est une puce sur laquelle sont combinés la matrice de cellules de la mémoire et les circuits de rafraîchissement. Extérieurement, cette puce s'utilise essentiellement comme une SRAM — on fournit l'adresse et on récupère la donnée; intérieurement toutefois, on retrouve une mémoire dynamique de haute densité. Le concepteur n'a plus à ajouter des circuits de rafraîchissement, qui sont intégrés à la logique interne de la puce; l'opération de rafraîchissement s'effectue automatiquement.

Ensuite, pour les systèmes de mémoire plus volumineux (> 64 kmots), la solution la plus rentable est de recourir à des puces LSI dites *contrôleurs de mémoire dynamique* qui renferme la totalité des circuits logiques qui ont la charge de rafraîchir les mémoires VIVES dynamiques se trouvant dans le système. Ceux-ci contribuent grandement à réduire l'importance des circuits additionnels d'un système DRAM.

Dans les applications où la vitesse et la simplicité priment sur l'encombrement et la consommation, le choix est simple, il faut prendre des SRAM. Ces mémoires sont généralement plus rapides que les mémoires VIVES dynamiques et n'ont pas besoin d'être rafraîchies. De conception plus simples, elles ne peuvent cependant rivaliser avec les mémoires VIVES dynamiques quand il s'agit d'obtenir une mémoire de grande capacité ayant une faible consommation.

À cause de la simplicité de la structure de leurs cellules, les DRAM occupent généralement 4 fois moins de superficie que les SRAM. C'est donc dire que sur la même surface on place 4 fois plus de cellules, ou ce qui est équivalent, il faut quatre fois moins de superficie pour obtenir une mémoire de capacité identique. Le coût par bit pour une mémoire VIVE dynamique est habituellement entre le cinquième et le quart de celui d'une mémoire statique. Une économie supplémentaire découle de la faible consommation des DRAM (entre le sixième et la moitié de celle des SRAM), soit l'utilisation de sources d'alimentation plus petites, donc, moins coûteuses.

Les principales applications dans lesquelles on retrouve des SRAM sont celles où la capacité de la mémoire n'est pas importante et où la vitesse est essentielle. De nombreux instruments et appareils électrodomestiques contrôlés par microprocesseurs n'ont pas besoin de beaucoup de mémoire. Également, de nombreux instruments, comme des oscilloscopes numériques à mémoire et des analyseurs logiques, réclament une mémoire très rapide. La SRAM convient donc tout particulièrement à ce genre d'applications.

On retrouve dans la mémoire interne de la plupart des micro-ordinateurs (IBM PC ou APPLE) de la mémoire DRAM, à cause des grandes capacités qu'on parvient à réaliser et de sa faible consommation. Ces ordinateurs ont toutefois recours à une petite quantité

de mémoire statique pour les fonctionnalités dont la réaction doit être très vive, comme les graphiques vidéo et les tables de consultation.

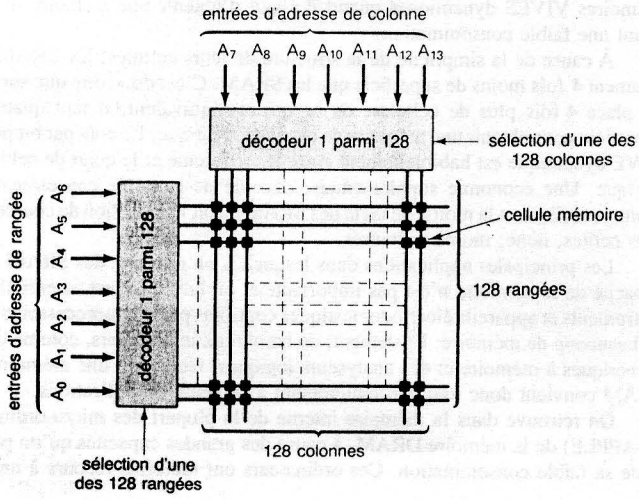
QUESTIONS DE RÉVISION

1. Donnez les principaux inconvénients de la mémoire DRAM par rapport à la mémoire SRAM.
2. Énumérez les avantages de la DRAM par rapport à la SRAM.
3. Quelles sont les deux solutions dont disposent les concepteurs de systèmes pour prendre en compte le rafraîchissement sans passer par l'addition de nombreux circuits additionnels.

11-14 STRUCTURE ET FONCTIONNEMENT DES MÉMOIRES VIVES DYNAMIQUES

La plupart des mémoires VIVES dynamiques peuvent être considérées comme une matrice de registres à 1 bit (cellule), comme celle de la figure 11-27. Dans cette figure, il y a 16 384 cellules agencées selon une matrice 128 × 128. Chaque cellule est repérée par une rangée particulière et une colonne particulière de la matrice. Quatorze entrées d'adresse sont nécessaires pour choisir une de ces cellules: les bits d'adresse de rangs inférieurs A_0–A_6 choisissent la rangée et les bits d'adresse des rangs supérieurs choisissent la colonne. Chaque adresse de 14 bits sélectionnent une cellule exclusive dont on lit la valeur ou dans laquelle on inscrit une valeur. La structure de la figure 11-27 est celle d'une puce de mémoire VIVE dynamique 16K × 1 (mémorisation de 16 384 mots de 1 bit). Il existe sur le marché des puces de mémoire VIVE dynamique dont les capacités peuvent atteindre 256K × 1. Les puces DRAM vendues actuellement dans le commerce ont des capacités de 1024K × 1 (1M × 1) et 256K × 4. Ces puces pour les mots de 4 bits ont une architecture semblable à celle présentée à la figure 11-27, à la seule différence qu'à la rencontre

FIGURE 11-27 Organisation de cellule dans une mémoire VIVE 16K × 1.



des lignes et des colonnes, il y a quatre cellules au lieu d'une seule. Chaque adresse sélectionne alors un groupe de quatre cellules pour une opération de lecture ou d'écriture. Comme nous le verrons plus loin, on parvient à constituer des mémoires pour des mots plus longs en combinant plusieurs puces selon une organisation particulière.

La figure 11-28 représente symboliquement une cellule de mémoire dynamique et ses circuits connexes. Malgré la simplicité et le manque de détails de ce schéma, il est suffisamment informatif pour que vous puissiez comprendre les principes essentiels mis en oeuvre durant une écriture et une lecture d'une DRAM. Les interrupteurs S1 à S4 sont en réalité des transistors MOS contrôlés par différentes sorties du décodeur d'adresse et par la ligne $L/\overline{E}$. Le condensateur est évidemment la cellule de mémorisation.

Pour écrire une donnée, les signaux des circuits logiques de décodage de l'adresse et de lecture/écriture ferment les interrupteurs S1 et S2, et gardent ouverts les interrupteurs S3 et S4. Cela raccorde la donnée d'entrée à C. Un 1 logique à l'entrée de donnée charge C, et un 0 logique le décharge. Ensuite, les interrupteurs sont ouverts de manière à déconnecter C du circuit.

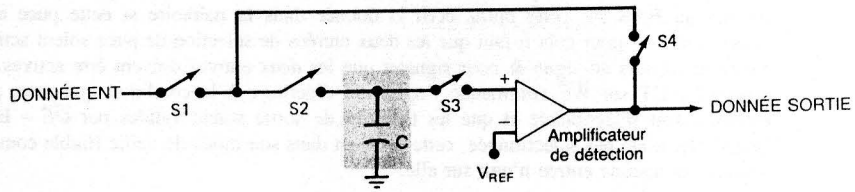


FIGURE 11-28 Représentation symbolique d'une cellule de mémoire dynamique. Durant une ÉCRITURE, les interrupteurs à semiconducteurs S1 et S2 sont fermés. Durant une LECTURE, ce sont les interrupteurs S3 et S4 qui sont fermés.

Idéalement, C devrait conserver sa décharge indéfiniment, mais des chemins de fuite au travers des interrupteurs fermés contribuent à l'épuisement progressif de la charge.

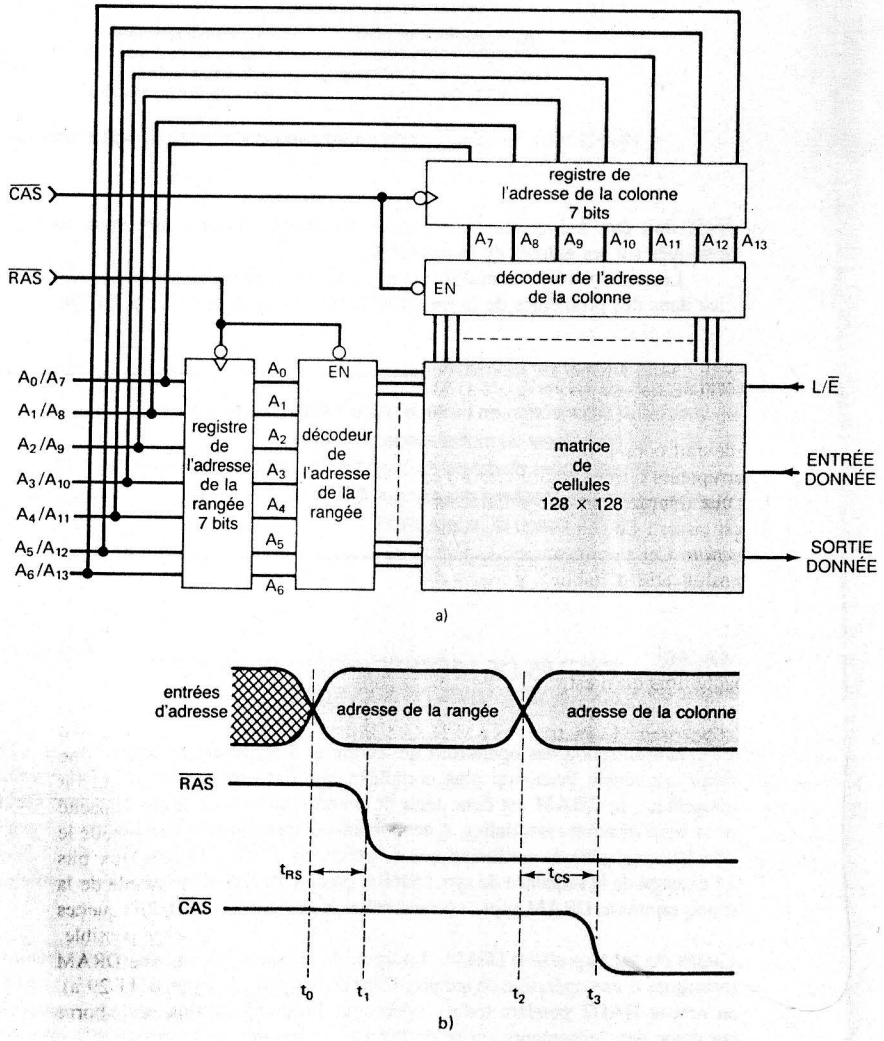
Pour lire une donnée dans une cellule, les interrupteurs S2, S3 et S4 sont fermés, tandis que S1 est ouvert. Le condensateur de mémorisation est alors raccordé à un *amplificateur de détection*. Cet amplificateur compare la tension du condensateur à une certaine référence de tension afin d'établir s'il s'agit d'un 0 ou d'un 1 logique; une fois cette détermination effectuée, il fournit sur la sortie de donnée un 0 V ou un 5 V. La sortie de donnée est également raccordée à C (S2 et S4 sont fermés) et elle rafraîchit le condensateur en le chargeant ou en le déchargeant. En d'autres mots, le bit de donnée dans la cellule est rafraîchi chaque fois qu'il est lu.

Multiplexage d'adresse La matrice DRAM 16K × 1 illustrée à la figure 11-27 possédait 14 entrées d'adresse. Une DRAM 64K × 1 aura besoin de 16 entrées d'adresse. Une mémoire de grande capacité nécessitera dès lors un grand nombre de bornes si chaque bit de l'adresse doit utiliser une broche. Les fabricants de puces DRAM de grande capacité ont trouvé une façon de réduire le nombre de bornes. Ils ont implanté ce qu'on appelle le *multiplexage d'adresse*; dans cette solution, chaque borne d'adresse accepte deux bits différents. La réduction du nombre de bornes se traduit par une diminution notable de la taille des boîtiers de CI. C'est une considération très importante dans le cas des plaquettes de mémoire de grande capacité sur lesquelles on veut loger le plus de mémoire possible.

Illustrons la notion du multiplexage d'adresse en nous servant du 4116, une DRAM de 16K × 1. On peut voir l'architecture simplifiée de cet élément à la figure 11-29 a). Remarquez la matrice de 128 × 128, une seule borne d'entrée de donnée, une seule borne

de sortie de donnée et une borne $L/\overline{E}$. On y trouve en outre sept entrées d'adresse, chacune jouant un double rôle (par ex. A_0/A_7 fonctionnent à la fois comme A_0 et A_7). Deux entrées de **strobage** de l'adresse sont incluses et ont pour fonction de commander l'introduction dans leur registre sur puce des adresses de la rangée et de la colonne. Le signal $\overline{RAS}$ **de strobage de l'adresse de la rangée** charge le registre d'adresse de la rangée ayant une longueur de 7 bits; le signal $\overline{CAS}$ **de strobage de l'adresse de la colonne** charge le registre d'adresse de la colonne, lui aussi d'une longueur de 7 bits.

FIGURE 11-29 a) Architecture simplifiée du 4116, une mémoire VIVE dynamique 16K × 1; b) chronogramme $\overline{RAS}$/$\overline{CAS}$.



Une adresse de 14 bits est appliquée à cette DRAM en deux phases au moyen des signaux $\overline{RAS}$ et $\overline{CAS}$. La synchronisation qui intervient est illustrée à la figure 11-29 b). Au début, $\overline{RAS}$ et $\overline{CAS}$ sont tous les deux au niveau HAUT. À l'instant t_0, l'adresse de rangée de 7 bits (A_0 à A_6) est appliquée sur les entrées d'adresse. Après un délai satisfaisant la condition d'établissement (t_{RS}) du registre de l'adresse de rangée, l'entrée $\overline{RAS}$ est forcée au niveau BAS en t_1. Ce front descendant charge l'adresse de rangée dans le registre du même nom et les valeurs sur A_0 et A_6 se trouvent désormais sur les entrées du décodeur de rangée. Le niveau BAS sur $\overline{RAS}$ valide aussi ce décodeur qui décode l'adresse de rangée et sélectionne une rangée dans la matrice.

À l'instant t_2, l'adresse de colonne de 7 bits (A_7 à A_{13}) est appliquée sur les entrées d'adresse. En t_3, l'entrée $\overline{CAS}$ passe au niveau BAS et commande l'introduction de l'adresse de colonne dans le registre du même nom. $\overline{CAS}$ valide au même instant le décodeur de colonne qui décode cette adresse et sélectionne une colonne dans la matrice.

À cette étape-ci, les deux portions de l'adresse se trouvent dans leur registre respectif. Les décodeurs ont sélectionné une cellule, soit celle qui est à la rencontre de la rangée et de la colonne décodées; une lecture ou une écriture utilisant cette cellule peut alors avoir lieu comme dans une mémoire VIVE statique.

Vous avez sans doute relevé que dans cette DRAM, il n'y a pas d'entrée de sélection de puce (CS). Les signaux $\overline{RAS}$ et $\overline{CAS}$ se chargent de réaliser la sélection de la puce étant donné qu'ils doivent tous les deux être au niveau BAS pour que les décodeurs sélectionnent une cellule dont on veut lire la valeur ou dans laquelle on veut écrire une valeur.

EXEMPLE 11-11

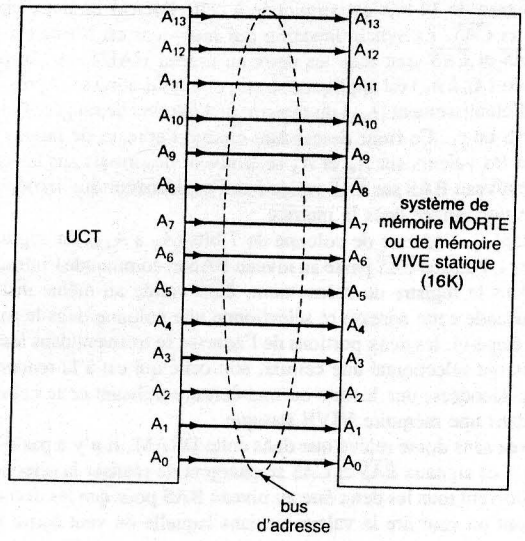
Combien de broches économise-t-on en recourant au multiplexage d'adresse dans une DRAM 16K × 1?

Solution On utilise sept bornes d'adresse au lieu de 14; comme il faut ajouter $\overline{RAS}$ et $\overline{CAS}$, mais qu'il n'y a plus de signal CS, l'économie nette est donc de *six* broches.

Dans un ordinateur type, les valeurs des entrées d'adresse de la mémoire proviennent de l'unité centrale de traitement (UCT). Quand cette unité veut accéder à un emplacement mémoire précis, elle génère l'adresse complète et la place sur les lignes d'adresse qui toutes ensemble constituent le *bus d'adresse*. La figure 11-30 a) illustre ceci pour le cas d'une mémoire ayant une capacité de 16K mots; il faut donc un bus d'adresse de 14 conducteurs qui relient directement l'UCT à la mémoire.

Ce montage fonctionne aussi bien pour les mémoires MORTES que pour les mémoires VIVES statiques, mais il faut le modifier dans le cas des mémoires VIVES dynamiques qui fonctionnent avec adressage multiplexé. Si la mémoire est une mémoire VIVE dynamique, elle n'aura que 7 entrées d'adresse. Cela implique que les 14 lignes d'adresse qui proviennent du bus d'adresse de l'UCT doivent arriver à un multiplexeur qui transmettra 7 bits d'adresse à la fois aux entrées d'adresse de la mémoire. C'est ce qui est schématisé sur la figure 11-30 b). L'entrée de sélection du multiplexeur, désignée *MUX*, donne l'ordre de placer sur les entrées d'adresse de la mémoire VIVE dynamique soit les lignes d'adresse de l'UCT A_0–A_6 ou les lignes d'adresse A_7–A_{13}.

Il est essentiel que le signal *MUX* soit synchronisé avec les signaux de strobage de l'adresse dans la mémoire $\overline{RAS}$ et $\overline{CAS}$. On peut voir à la figure 11-31 la synchronisation qu'il faut respecter: *MUX* doit être à 0 quand $\overline{RAS}$ passe à 0 de façon à ce que les lignes d'adresse A_0–A_6 provenant de l'UCT soient placées sur les entrées d'adresse de la mémoire et envoyées dans le registre à mémoire au moment du front descendant de $\overline{RAS}$. De même,



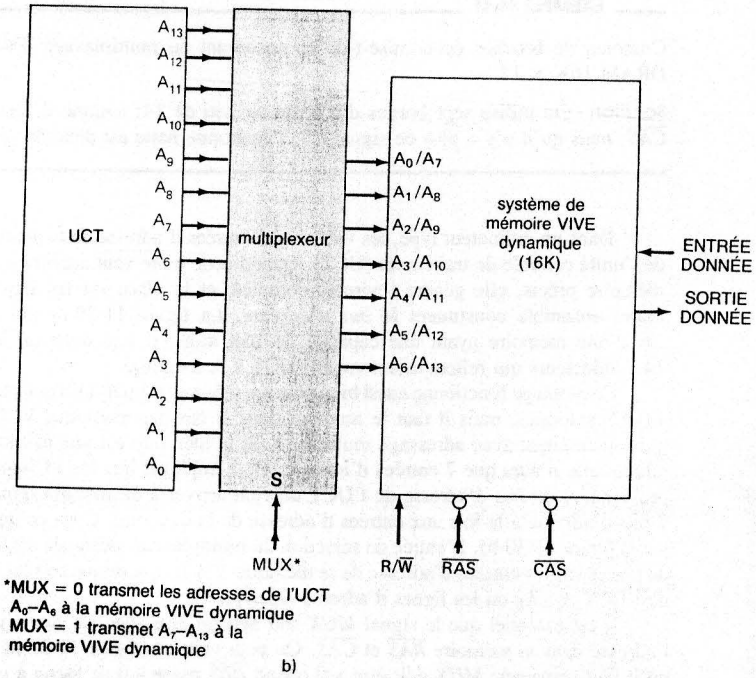


FIGURE 11-30 a) Bus d'adresse de l'UCT pilotant une mémoire MORTE ou une mémoire VIVE statique; b) adresses de l'UCT pilotant une mémoire VIVE dynamique par l'entremise d'un multiplexeur qui choisit les lignes d'adresses de l'UCT.

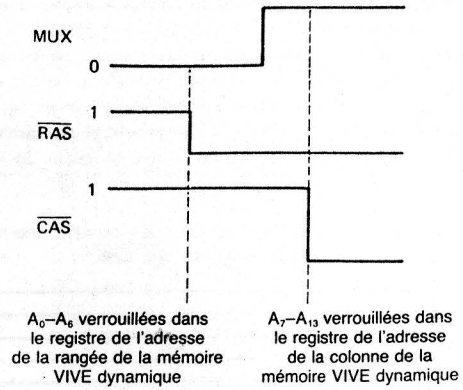


FIGURE 11-31 Synchronisation nécessaire pour le multiplexage d'adresse.

MUX doit être à 1 quand $\overline{CAS}$ passe à 0 afin que A_7–A_{13} provenant de l'UCT soient appliqués sur les entrées de la mémoire.

Les circuits réels de multiplexage et de synchronisation seront abordés à titre d'exercice dans des problèmes de la fin du chapitre. (Problèmes 11-25 et 11-26).

QUESTIONS DE RÉVISION

1. Décrivez l'organisation en matrice d'une DRAM de 64K × 1.
2. Quel est l'avantage du multiplexage d'adresse?
3. Combien d'entrées d'adresse faudrait-il pour une puce DRAM de 1M × 1?
4. Quelles sont les fonctions des signaux $\overline{RAS}$ et $\overline{CAS}$?
5. Quelle est la fonction du signal *MUX*?

11-15 CYCLES DE LECTURE/ÉCRITURE D'UNE MÉMOIRE VIVE DYNAMIQUE

La synchronisation des opérations de lecture et d'écriture dans une mémoire VIVE dynamique se révèle beaucoup plus complexe que dans une mémoire VIVE statique, et le concepteur de DRAM est donc tenu de prendre en considération plusieurs spécifications de synchronisation essentielles. À ce moment-ci, un traitement en détail de ces spécifications pourrait créer plus de confusion que de précision. C'est pourquoi nous allons nous limiter à l'examen de la séquence de synchronisation de base des opérations de lecture et d'écriture d'une mémoire DRAM type, comme celle de la figure 11-30 b).

Cycle de lecture d'une DRAM La figure 11-32 nous fait voir les chronogrammes caractéristiques d'une opération de lecture. Dans cette figure, le signal $L/\overline{E}$ (non illustré) demeure au niveau HAUT pendant toute l'opération. La partie suivante est une description étape par étape des événements qui se déroulent aux instants notés sur ces chronogrammes.

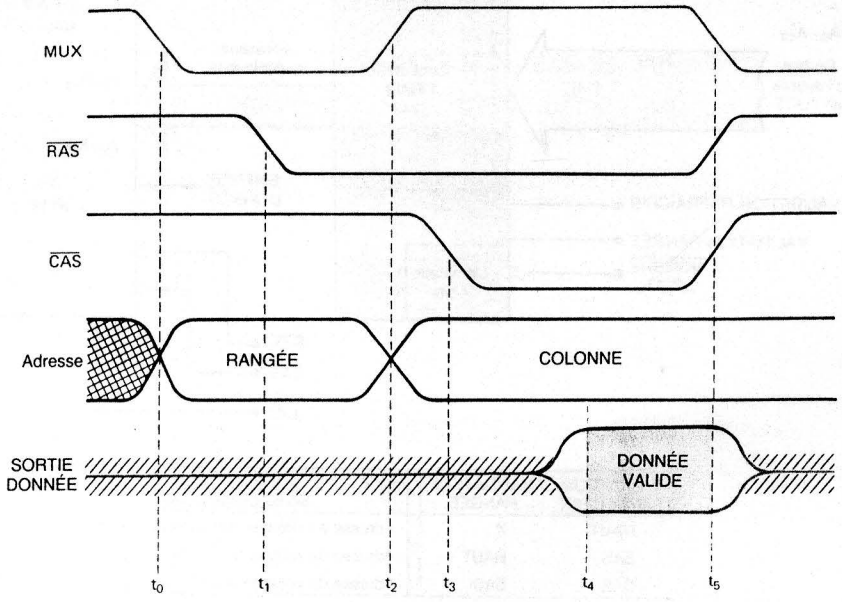


FIGURE 11-32 Chronogrammes des signaux qui sont activés au moment d'une opération de lecture d'une mémoire VIVE dynamique. L'entrée $L/\overline{E}$ (non montrée) est supposée au niveau HAUT.

t_0: *MUX* est porté au niveau BAS afin de placer les bits d'adresse de la rangée (A_0 à A_6) sur les entrées d'adresse de la DRAM.

t_1: $\overline{RAS}$ est amené au niveau BAS afin de mémoriser l'adresse de la rangée dans la DRAM.

t_2: *MUX* effectue une transition vers le niveau HAUT afin de placer l'adresse de la colonne (A_7 à A_{13}) sur les entrées d'adresse de la DRAM.

t_3: $\overline{CAS}$ passe au niveau BAS afin de mémoriser l'adresse de la colonne dans la DRAM.

t_4: La DRAM répond ensuite en plaçant sur la ligne de SORTIE DONNÉE la donnée valide se trouvant dans la cellule adressée.

t_5: *MUX*, $\overline{RAS}$, $\overline{CAS}$ et SORTIE DONNÉE reviennent à leur état initial.

Cycle d'écriture d'une DRAM La figure 11-33 nous montre les chronogrammes caractéristiques d'une opération d'écriture dans une mémoire VIVE dynamique. Voici la description détaillée de ce qui se produit.

t_0: Le niveau BAS sur *MUX* place l'adresse de la rangée sur les entrées de la DRAM.

t_1: Le front descendant de $\overline{RAS}$ déclenche le placement de l'adresse de la rangée dans la DRAM.

t_2: *MUX* passe au niveau HAUT et applique l'adresse de la colonne sur les entrées de la DRAM.

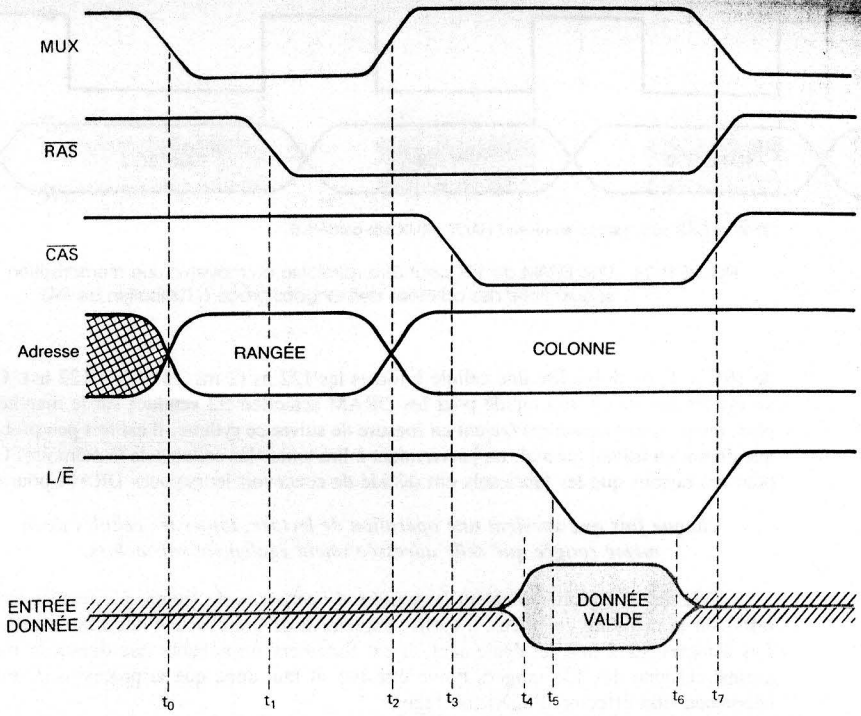


FIGURE 11-33 Chronogrammes des signaux qui sont activés durant une opération d'écriture d'une mémoire VIVE dynamique.

t_3: Le front descendant de $\overline{CAS}$ mémorise l'adresse de la colonne dans la DRAM.

t_4: La donnée à écrire est placée sur la ligne ENTRÉE DONNÉE.

t_5: $L/\overline{E}$ est mis au niveau BAS en vue d'écrire la donnée dans la cellule adressée.

t_6: La donnée entrée est enlevée de la ligne ENTRÉE DONNÉE.

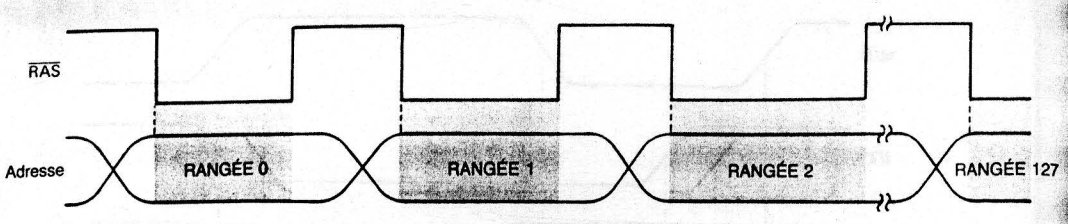
t_7: *MUX*, $\overline{RAS}$, $\overline{CAS}$ et $L/\overline{E}$ reviennent à leur état initial.

QUESTIONS DE RÉVISION

1. Énumérez les étapes intervenant dans une opération de lecture d'une DRAM.
2. Énumérez les étapes intervenant dans une opération d'écriture.

11-16 RAFRAÎCHISSEMENT DES DRAM

Les cellules de DRAM doivent être rafraîchies périodiquement (généralement à toutes les 2 ms), sans quoi les données qui y sont mémorisées seront perdues. Comme nous l'avons expliqué dans notre discussion de la DRAM de la figure 11-28, une cellule est rafraîchie chaque fois qu'elle est lue. Il semble donc que pour rafraîchir une mémoire VIVE dynamique



*R/$\overline{W}$ et $\overline{CAS}$ sont gardés au niveau HAUT; MUX est gardé à 0.

FIGURE 11-34 Une DRAM de 16K peut être rafraîchie au moyen d'une mémorisation séquentielle des adresses des rangées grâce à l'utilisation de *RAS*.

de 16K × 1, on doive lire une cellule à toutes les 122 ns (2 ms/16 384 = 122 ns). C'est un rythme beaucoup trop rapide pour les DRAM actuellement vendues sur le marché; en plus, même si ces mémoires étaient en mesure de suivre ce rythme, il est fort peu probable que durant un travail normal, on parviennent à lire toutes les cellules de la mémoire. C'est pour ces raisons que les fabricants ont décidé de concevoir leurs puces DRAM pour que:

chaque fois que survient une opération de lecture, toutes les cellules de la même rangée que celle adressée soient également rafraîchies.

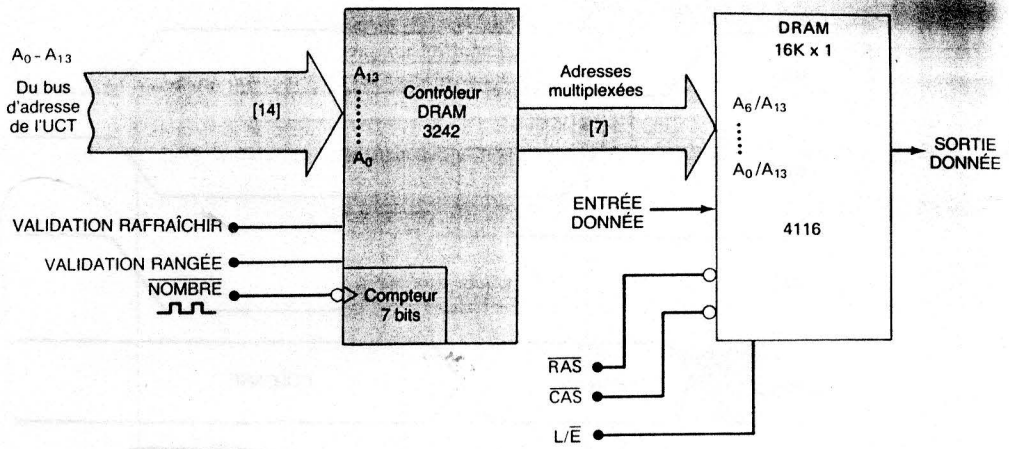
Cela réduit considérablement le nombre d'opérations de lecture à effectuer afin de rafraîchir la mémoire complète; il suffit simplement de lire chacune des 128 rangées une fois à toutes les 2 ms. Ici également, il est fortement improbable que durant le travail normal chacune des 128 rangées doive être lue. Il faut donc que le processus de rafraîchissement soit effectué d'une autre façon.

La solution la plus courante est d'associer à la mémoire un **compteur de rafraîchissement** de 7 bits qui parcourt cycliquement les 128 adresses différentes pour les rangées. Le compteur a comme valeur de départ 0000000, qui est l'adresse de la rangée 0. Cette adresse est appliquée aux entrées d'adresse de la DRAM ($MUX = 0$), $\overline{RAS}$ est changé pour le niveau BAS (voyez figure 11-34), pendant que $L/\overline{E}$ et $\overline{CAS}$ sont gardés au niveau HAUT. Ces conditions ont pour effet de rafraîchir la rangée 0. Le compteur est incrémenté et la séquence est répétée jusqu'à la rangée 127. Cela met un terme au processus de rafraîchissement qui a été effectué en 50 μs approximativement.

Comme tel, le compteur de rafraîchissement est une idée assez simple, c'est toutefois oublier que les adresses des rangées qu'il envoie ne doivent jamais interférer avec les adresses en provenance de l'UCT et destinées à des opérations de lecture et d'écriture. Pour cette raison, les adresses du compteur de rafraîchissement doivent être multiplexées avec celles de l'UCT, afin que la bonne source d'adresses de la DRAM soit activée au moment opportun.

La majorité des fabricants de mémoires dynamiques intégrés ont mis au point des CI spéciaux qui gèrent le rafraîchissement ainsi que le multiplexage des adresses indispensables aux systèmes DRAM. Ces CI s'appellent des **contrôleurs de RAM dynamiques**. Nous allons examiner une puce type de ce genre, l'Intel 3242, conçue pour les DRAM 16K. Le schéma fonctionnel de ce contrôleur est reproduit à la figure 11-35.

Le 3342 fournit en sortie une adresse multiplexée de 7 bits qui aboutit sur les entrées d'adresse de la DRAM. Cette adresse peut provenir d'une source parmi trois sources possibles qui sont: le compteur de rafraîchissement de 7 bits, l'adresse de rangée de l'UCT et l'adresse de colonne de l'UCT. Le compteur est incrémenté par une horloge externe sur sa borne $\overline{NOMBRE}$. Ce compteur fournit les adresses des rangées de la DRAM pendant



VALIDATION RAFRAÎCHIR	VALIDATION RANGÉE	Sortie du contrôleur
HAUT	X	Adresse à rafraîchir (du compteur interne)
BAS	HAUT	Adresse de rangée (A_0–A_6 de l'UCT)
BAS	BAS	Adresse de colonne (A_7–A_{13} de l'UCT)

FIGURE 11-35 Le contrôleur de DRAM 3242 se charge du multiplexage de l'adresse et du comptage de l'adresse à rafraîchir d'une mémoire de 16K.

un processus de rafraîchissement. Le 3242 reçoit en outre l'adresse de 14 bits du bus de l'UCT et la divise en une adresse de rangée et une adresse de colonne quand l'UCT procède à une opération de lecture ou d'écriture. Les niveaux logiques placées sur les bornes VALIDATION RAFRAÎCHIR et VALIDATION RANGÉE stipulent quelle adresse de 7 bits doit aboutir jusqu'aux sorties du contrôleur. La table de décision est reproduite sur cette figure.

QUESTIONS DE RÉVISION

1. Quel est le rôle du compteur de rafraîchissement?
2. Quelles fonctions sont réalisées par le contrôleur de DRAM?

11-17 EXTENSION DE LA LONGUEUR DU MOT ET DE LA CAPACITÉ

Dans la plupart des situations où on utilise des mémoires intégrées, la capacité et la longueur du mot d'une seule puce mémoire sont rarement suffisantes. C'est pourquoi on doit assembler plusieurs puces mémoires pour obtenir la capacité et la longueur de mot recherchées. Dans la présente section, nous verrons comment on y parvient en étudiant quelques exemples qui illustrent toutes les notions importantes que l'on doit savoir quand on interface des puces mémoires avec un microprocesseur.

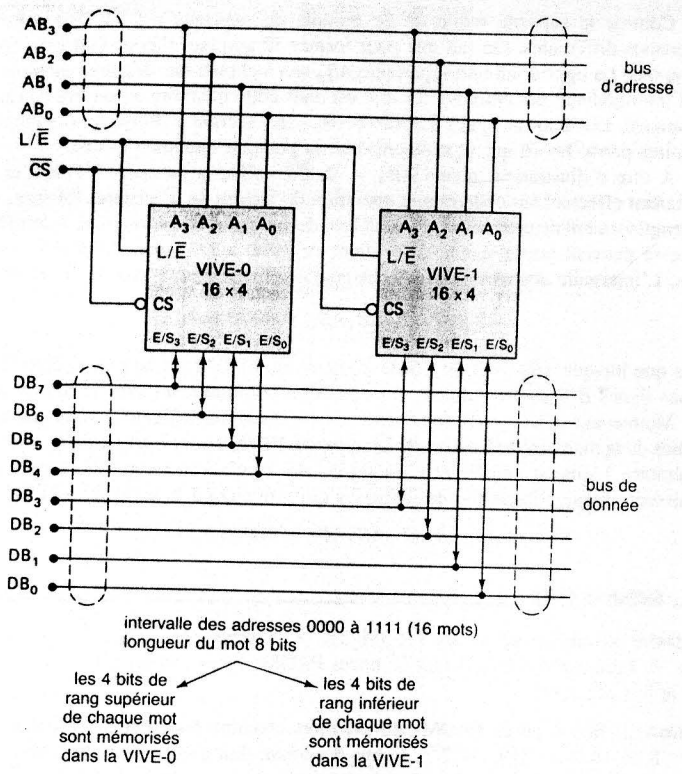


FIGURE 11-36 Assemblage de deux mémoires VIVES 16 × 4 pour constituer une mémoire 16 × 8.

Extension de la longueur du mot Supposons que nous ayons besoin d'une mémoire qui peut emmagasiner 16 mots de 8 bits de longueur et que nous ayons à notre disposition des puces de mémoire VIVE qui sont des matrices 16 × 4 ayant des lignes E/S communes. Il est possible d'assembler deux de ces puces 16 × 4 pour obtenir la mémoire recherchée. On réussit cet assemblage en configurant les systèmes comme à la figure 11-36. Examinez bien ce schéma et voyez ce que vous pouvez en déduire avant de lire le texte qui suit.

Étant donné que chaque puce peut mémoriser 16 mots de 4 bits et que l'on en veut 16 de 8 bits, nous mémoriserons dans chaque puce la *moitié* de chaque mot. Autrement dit, la mémoire VIVE 0 mémorise les 4 bits de rang *supérieur* de chacun des 16 mots, et la mémoire VIVE 1 mémorise les 4 bits de rang *inférieur* de chacun de ces 16 mots. Un mot d'une longueur totale de 8 bits se retrouve donc sur les sorties des mémoires VIVES connectées au bus de donnée.

N'importe lequel de ces 16 mots est choisi en appliquant l'adresse appropriée au *bus d'adresse* de 4 conducteurs (AB_3, AB_2, AB_1, AB_0). Habituellement, les lignes d'adresse proviennent de l'UCT de l'ordinateur. Notez que chacun des conducteurs du bus d'adresse est branché sur l'entrée d'adresse correspondante de chaque puce. Cela veut donc dire que chaque fois qu'une adresse est placée sur le bus d'adresse, celle-ci est appliquée aux deux puces, de sorte que les emplacements homologues (qui se correspondent) de chaque puce sont adressés en même temps.

Dès qu'une adresse a choisi un registre, on peut lire ou on peut écrire dans l'emplacement adressé suivant les valeurs sur les lignes $L/\overline{E}$ et $\overline{CS}$. Pour lire, $L/\overline{E}$ doit être à 1 et $\overline{CS}$ à 0. Ceci a pour effet de faire agir les lignes E/S de la mémoire VIVE comme des *sorties*. La mémoire VIVE-0 place son mot de 4 bits choisi sur les 4 conducteurs supérieurs du bus de donnée et la mémoire VIVE-1 place son mot de 4 bits choisi sur les 4 conducteurs inférieurs de ce bus. Le bus de donnée renferme alors le mot choisi d'une longueur totale de 8 bits, qui peut maintenant être transmis à un autre dispositif (par exemple un registre de l'UCT).

Pour écrire, $L/\overline{E} = 0$ et $\overline{CS} = 0$, ce qui transforme les lignes E/S des mémoires VIVES en *entrées*. Le mot de 8 bits à inscrire est placé sur le bus de donnée par un dispositif extérieur. Les 4 bits supérieurs sont inscrits dans l'emplacement choisi de la mémoire VIVE-0 et les 4 bits inférieurs sont inscrits dans l'emplacement de la mémoire VIVE-1.

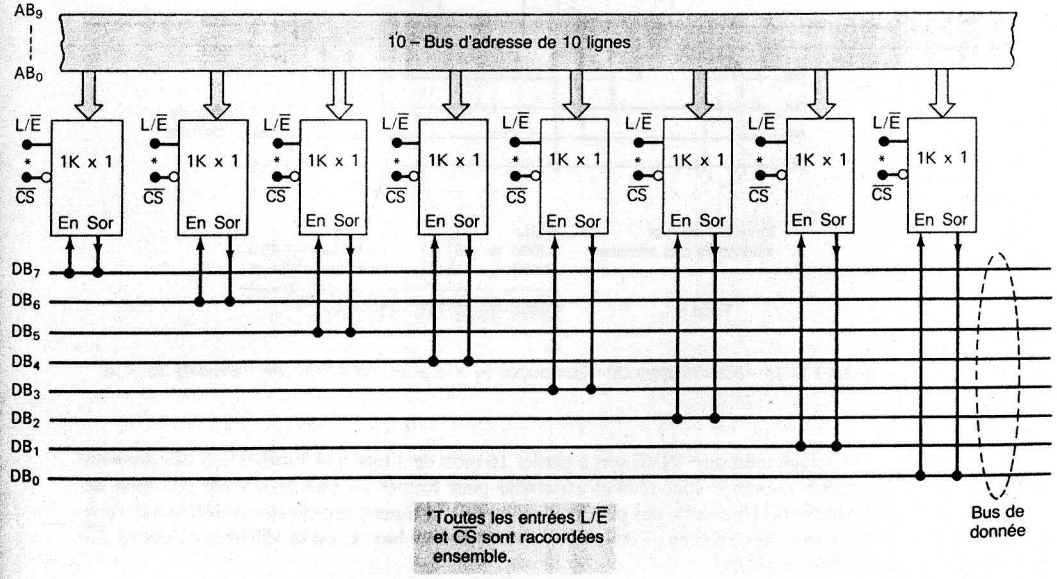
Essentiellement, l'assemblage de deux puces de mémoire VIVE est identique à une seule puce mémoire 16 × 8. Nous donnons à un tel assemblage le nom de module mémoire 16 × 8.

Le même principe d'extension de la longueur d'un mot peut être étendu à d'autres situations différentes. Lisez l'énoncé de l'exemple suivant et tracez grossièrement le schéma de la solution.

EXEMPLE 11-12

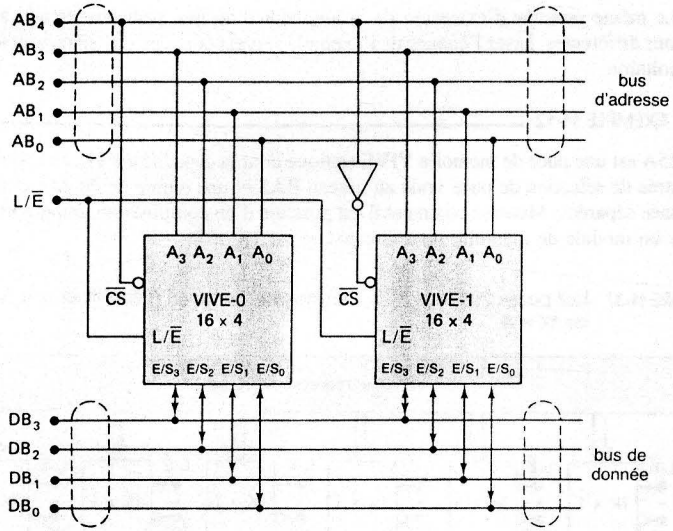
Le 2125A est une puce de mémoire VIVE statique dont la capacité est 1K × 1, qui possède une entrée de sélection de puce vraie au niveau BAS et une entrée de donnée et une sortie de donnée séparées. Montrez comment il est possible d'en combiner un grand nombre pour obtenir un module de mémoire dont la capacité est 1K × 8.

FIGURE 11-37 Huit puces 2125A de 1K × 1 organisées de façon à constituer une mémoire de 1K × 8.



Solution Le montage nécessaire est illustré à la figure 11-37 où l'on peut voir huit puces 2125A agencées afin de réaliser une mémoire de 1K × 8. Chaque puce mémorise un bit de chacun des 1024 mots de huit bits. Remarquez que les entrées $L/\overline{E}$ et $\overline{CS}$ sont raccordées ensemble et que la ligne d'adresse de 10 bits est connectée aux entrées d'adresse de chaque puce. Observez en outre les broches séparées sur chaque 2125A pour les entrées et les sorties; pour une même puce, ces broches sont reliées au même conducteur du bus de donnée.

Extension de la capacité Supposons que l'on ait besoin d'une mémoire de 32 mots de 4 bits et que l'on ne dispose que de puces de 16 × 4. Il est possible en assemblant deux puces 16 × 4 comme à la figure 11-38 de construire la mémoire recherchée. À nouveau, examinez ce diagramme et trouvez ce que vous pouvez en déduire avant de lire le texte qui suit.



longueur du mot	4 bits		
intervalles des adresses	00000 à	01111	– mots dans VIVE-0
	10000 à	11111	– mots dans VIVE-1
Total →	00000 à	11111	– 32 mots

FIGURE 11-38 Assemblage de deux puces 16 × 4 pour constituer une mémoire 32 × 4.

Chaque mémoire VIVE sert à garder 16 mots de 4 bits. Les broches E/S des données de chaque mémoire sont réunies ensemble pour former un bus de donnée commun de 4 conducteurs. Une seule des puces de mémoire VIVE peut être choisie (validée) à la fois, ainsi on évite les problèmes de conflit d'utilisation du bus. C'est la valeur sur l'entrée $\overline{CS}$ qui établit ce choix.

Comme la capacité totale de ce module de mémoire est 32 X 4, on doit avoir 32 adresses différentes. On sait que pour former 32 adresses il nous faut *cinq* conducteurs dans le bus. La ligne d'adresse supérieure AB_4 sert à choisir une des deux mémoires VIVES (par l'intermédiaire des entrées $\overline{CS}$); et c'est dans cette mémoire qu'on lira ou qu'on écrira une donnée. Les 4 autres lignes d'adresse AB_0–AB_3 servent à choisir un des emplacements mémoires parmi les 16 qui se trouvent dans la puce de mémoire VIVE.

À titre d'illustration, quand $AB_4 = 0$, $\overline{CS}$ valide la mémoire VIVE-0 et l'on peut maintenant effectuer sur celle-ci une opération de lecture ou d'écriture. Ensuite, n'importe quel emplacement de cette mémoire peut être désigné par AB_3–AB_0. Ces 4 dernières lignes d'adresse peuvent porter des valeurs allant de 0000 à 1111 pour choisir l'emplacement voulu. L'intervalle des adresses pour les emplacements de la mémoire VIVE-0 est

$$AB_4AB_3AB_2AB_1AB_0 = 00000 \text{ à } 01111$$

Notez que lorsque $AB_4 = 0$, le $\overline{CS}$ de la mémoire VIVE-1 est au niveau HAUT, de sorte que ses lignes E/S sont invalidées et ne peuvent communiquer avec le bus de donnée.

Maintenant si $AB_4 = 1$, vous ne devez avoir aucune difficulté à vous convaincre que les rôles de la mémoire VIVE-0 et de la mémoire VIVE-1 sont intervertis. C'est maintenant la mémoire 1 qui est validée et c'est un de ses emplacements que les lignes AB_3–AB_0 choisissent. Ainsi, l'intervalle des adresses qui renvoient à la mémoire VIVE-1 est:

$$AB_4AB_3AB_2AB_1AB_0 = 10000 \text{ à } 11111$$

EXEMPLE 11-13

On désire assembler plusieurs PROM 2K × 8 pour obtenir une capacité totale de 8K × 8. Indiquez combien il faut de puces PROM. Dites combien il faut de conducteurs dans le bus d'adresse.

Solution Il faut 4 puces PROM, de 2K mots chacune pour obtenir 8K mots. Puisque $8K = 8 \times 1024 = 8192 = 2^{13}$, le bus d'adresse doit avoir 13 conducteurs.

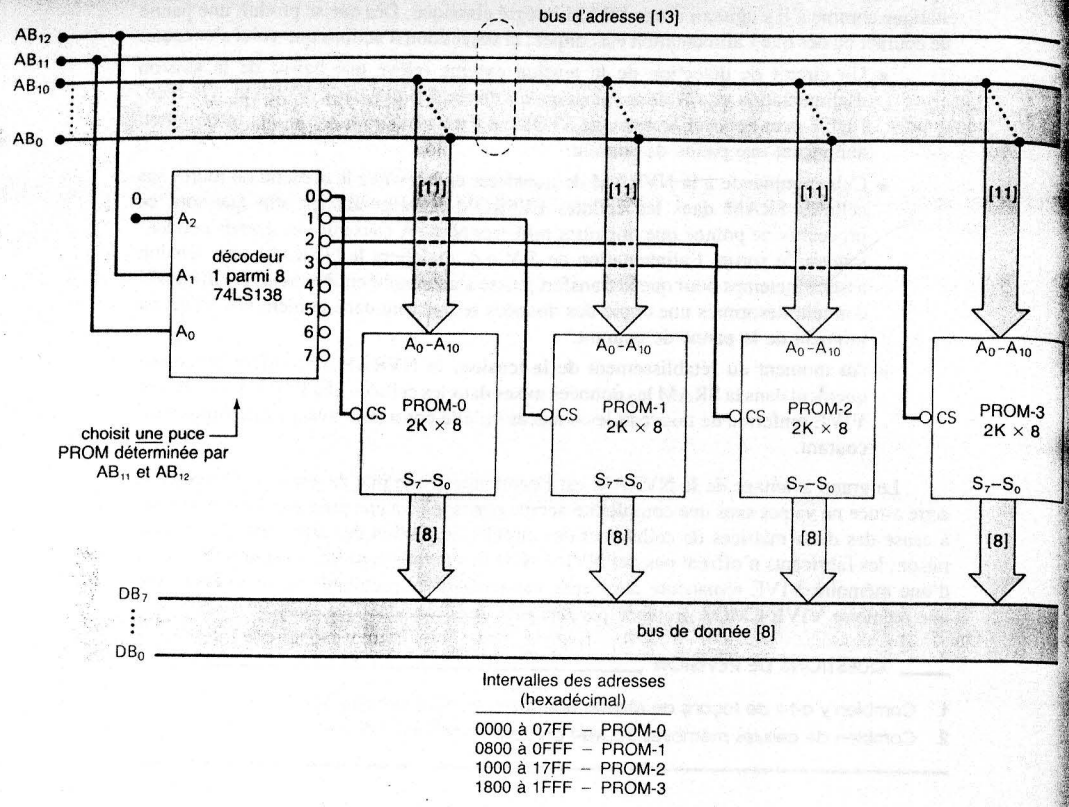
La configuration de la mémoire de l'exemple 11-13 est similaire à celle de la mémoire 32 × 4 de la figure 11-38. Toutefois, elle est légèrement plus complexe, puisqu'il faut un circuit de décodage pour produire les signaux d'entrée $\overline{CS}$. Vous pouvez voir le schéma complet de la mémoire 8K × 8 à la figure 11-39.

Comme la capacité est 8192 mots, il faut, comme on l'a calculé, 13 conducteurs dans le bus d'adresse. Les deux lignes de rang supérieur, AB_{11} et AB_{12}, servent à choisir *une* des puces PROM. Les autres 11 conducteurs du bus d'adresse aboutissent à chaque PROM pour choisir l'emplacement désiré dans la PROM sélectionnée. La sélection de la PROM se fait en appliquant AB_{11} et AB_{12} au décodeur. Les quatre combinaisons possibles de AB_{11}, AB_{12} sont décodées pour donner des signaux vrais au niveau BAS, qui sont ensuite appliqués aux entrées $\overline{CS}$. Par exemple, quand $AB_{11} = AB_{12} = 0$, la sortie 0 du décodeur passe au niveau BAS (tandis que toutes les autres restent au niveau HAUT) et valide la PROM-0. Il apparaît alors sur les sorties de cette PROM la donnée mémorisée à l'adresse repérée par AB_0–AB_{10}. Toutes les autres PROM sont invalidées.

Toutes les adresses de l'intervalle que voici concernent la PROM-0:

$$AB_{12}\text{–}AB_0 = 0000000000000 \text{ à } 0011111111111$$

Il s'agit des 2048 premières adresses de la mémoire. Pour plus de commodité, il est préférable d'écrire ces adresses en hexadécimal, ce qui donne l'intervalle 0000_{16} à $07FF_{16}$.



Intervalles des adresses (hexadécimal)

0000 à 07FF – PROM-0
0800 à 0FFF – PROM-1
1000 à 17FF – PROM-2
1800 à 1FFF – PROM-3

FIGURE 11-39 PROM 8K × 8 formée au moyen de 4 puces 2K × 8.

De même, $AB'_{12} = 0$ et $AB_{11} = 1$ choisissent la PROM-1 dont l'intervalle des adresses est

0100000000000 à 0111111111111 (binaire)

ou

0800 à 0FFF (hexadécimal)

Nous laissons au lecteur le soin de vérifier que les intervalles des adresses des PROM-2 et PROM-3 sont ceux donnés à la figure 11-39.

EXEMPLE 11-14

Dites quelles caractéristiques doit avoir un décodeur pour qu'on puisse étendre la mémoire de la figure 11-39 jusqu'à une 32K × 8. Indiquez le nombre de conducteurs nécessaires pour le bus d'adresse.

Solution Une capacité de 32K implique l'utilisation de 16 puces PROM. Pour choisir une puce parmi 16, il faut utiliser un décodeur 1 parmi 16. En plus, une capacité de 32 768 mots exige un total de 15 lignes d'adresse: 4 pour aller au décodeur afin de choisir la PROM et les autres 11 aboutissant à chaque PROM pour choisir l'adresse d'un emplacement dans celle-ci.

Combinaison de DRAM Comme nous l'avons déjà dit, les DRAM intégrées que l'on trouve sur le marché ont des mots de 1 ou 4 bits. Pour les incorporer dans les ordinateurs dont les mots ont des longueurs de 8 ou 16 bits, il est indispensable d'en combiner plusieurs selon une architecture semblable à celle suivie pour la combinaison des mémoires VIVES et MORTES statiques. La figure 11-40 nous montre la façon de combiner huit DRAM 16K × 1 afin d'obtenir un module de mémoire 16K × 8. Le contrôleur DRAM 3242 sert au multiplexage des adresses et à la fonction de rafraîchissement de chacune des puces.

QUESTIONS DE RÉVISION

1. La 2114 est une puce de mémoire VIVE statique de 1K × 4. Combien faut-il en utiliser pour constituer un module de 16K × 4?
2. Combien faut-il utiliser de 2114 pour constituer un module de 1K × 16?
3. *Vrai ou faux:* Quand on combine des puces mémoires afin d'étendre la capacité ou la longueur du mot, les entrées *CS* de chaque puce sont toujours raccordées ensemble.
4. *Vrai ou faux:* Quand on combine des puces mémoires afin d'étendre la capacité, chaque puce est reliée aux mêmes conducteurs du bus de donnée.

11-18 MÉMOIRE VIVE RÉMANENTE

Les mémoires VIVES à semiconducteurs disposent d'un avantage certain dans les opérations rapides; cependant, elles sont volatiles, c'est-à-dire que leur contenu disparaît dès que la tension est enlevée, même un court instant. Les mémoires MORTES, par contre, sont rémanentes, mais il est impossible de s'en servir comme mémoires à lecture/écriture. Dans certaines applications, la volatilité des mémoires VIVES peut entraîner la perte de données importantes ou cruciales si une panne de courant survient. Face à ce problème, il existe deux solutions. D'abord, recourir à une mémoire alimentée par des piles de secours qui entrent en action dès que survient une panne. Une telle mémoire consomme très peu d'énergie, de sorte que la capacité des piles n'est pas épuisée trop rapidement. Ce sont les CMOS qui, parmi toutes les mémoires VIVES, sont les moins gourmandes en énergie et qui peuvent, la plupart du temps, être alimentées par des piles de secours. Évidemment, quand elles tirent leur énergie des piles de secours, ces mémoires VIVES CMOS sont dans leur état d'attente (de veille) afin de faire durer les piles le plus longtemps possible. Certaines SRAM CMOS contiennent directement dans leur boîtier une petite pile de secours au silicium.

L'autre solution est de se servir d'un élément appelé mémoire VIVE *rémanente* (*NVRAM*). Une NVRAM renferme une matrice de mémoires VIVES statiques et une matrice d'EEPROM (PROM effaçable électriquement) sous le même boîtier. Cet élément combine la rapidité de lecture et d'écriture des mémoires VIVES statiques et la rémanence des EEPROM. Chaque cellule de mémoire statique est dédoublée par une cellule EEPROM et les données sont transférées, dans les deux sens, de l'une vers l'autre. C'est ainsi qu'en fonctionnement normal, les opérations d'écriture et de lecture se passent dans la mémoire

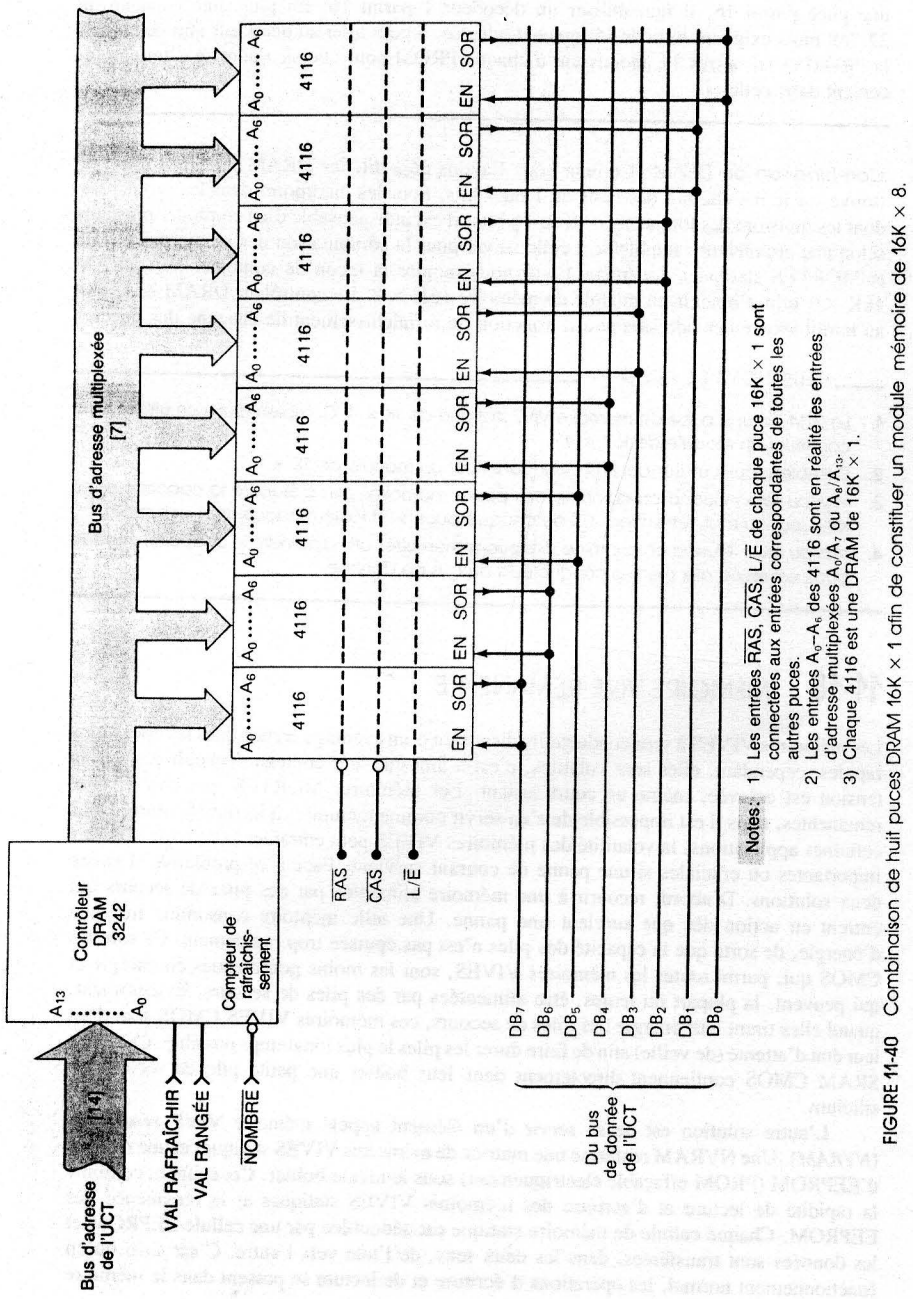


FIGURE 11-40 Combinaison de huit puces DRAM 16K × 1 afin de constituer un module mémoire de 16K × 8.

statique comme s'il s'agissait d'une SRAM intégré classique. Dès que se produit une panne de courant ou dès que l'alimentation est coupée, la succession d'actions que voici s'exécute:

- Un circuit de détection de la tension externe relève une baisse de la tension d'alimentation et envoie un message à l'entrée STORE de la NVRAM. Ou bien, l'UCT peut envoyer le message STORE à l'instant où elle reçoit une interruption annonçant une panne de courant.
- Cela commande à la NVRAM de transférer en *parallèle* le contenu de toutes ses cellules SRAM dans les cellules EEPROM correspondantes afin que tout ce processus ne prenne que quelques millisecondes. À cause de ses grands condensateurs de sortie, l'alimentation de 5 V c.c. maintient la NVRAM sous tension assez longtemps pour que le transfert puisse s'accomplir entièrement. L'EEPROM contient désormais une copie des données se trouvant dans la mémoire VIVE au moment de la panne de courant.
- Au moment du rétablissement de la tension, la NVRAM retransfère automatiquement dans la SRAM les données mises dans les cellules EEPROM. La mémoire VIVE renferme de nouveau les données qu'elle conservait avant l'interruption du courant.

Le grand avantage de la NVRAM est l'économie d'une pile de secours. Cependant, cette astuce ne va pas sans une complexité accrue par rapport à une puce mémoire normale, à cause des deux matrices de cellules et des circuits de gestion des transferts. Pour cette raison, les fabricants n'offrent pas des NVRAM de très grande capacité. Quand on a besoin d'une mémoire VIVE rémanente de grande capacité, il est préférable de se tourner vers une mémoire VIVE CMOS protégée par une pile de secours.

QUESTIONS DE RÉVISION

1. Combien y a-t-il de façons de réaliser une mémoire VIVE rémanente?
2. Combien de cellules mémoires trouve-t-on dans un NVRAM 512 × 8?

11-19 MÉMOIRES SÉQUENTIELLES

Les mémoires à semiconducteurs discutées jusqu'à présent étaient du type à accès sélectif, puisque une donnée dans un emplacement quelconque en mémoire est récupérée directement et rapidement sans l'examen d'aucun autre emplacement. Le fonctionnement très rapide des mémoires à accès sélectif les destine tout particulièrement à servir comme mémoire interne dans les ordinateurs. Les mémoires à semiconducteurs à accès séquentiel font usage de registres à décalage pour le stockage des données qui seront récupérées de manière séquentielle. Bien qu'inutilisables comme mémoires internes dans les ordinateurs à cause de leur lenteur, les mémoires à registres à décalage trouvent un champ d'application dans les appareils exigeant des données séquentielles et répétitives. Un exemple immédiat est la mémorisation et la transmission des données ASCII qui correspondent aux caractères apparaissant sur les écrans de visualisation. Ces données sont fournies régulièrement aux circuits d'affichage vidéo afin de regénérer l'image reproduite à l'écran. Au moyen de registres à décalage, les données en mémoire sont *recirculées* sans cesse afin de regénérer l'image à l'écran. On retrouve également des mémoires séquentielles dans les oscilloscopes numériques à mémoire et les analyseurs logiques.

Registre à décalage circulaire La figure 11-41 a) représente le schéma fonctionnel d'un registre à décalage circulaire. Il peut y avoir un nombre quelconque de bascules dans

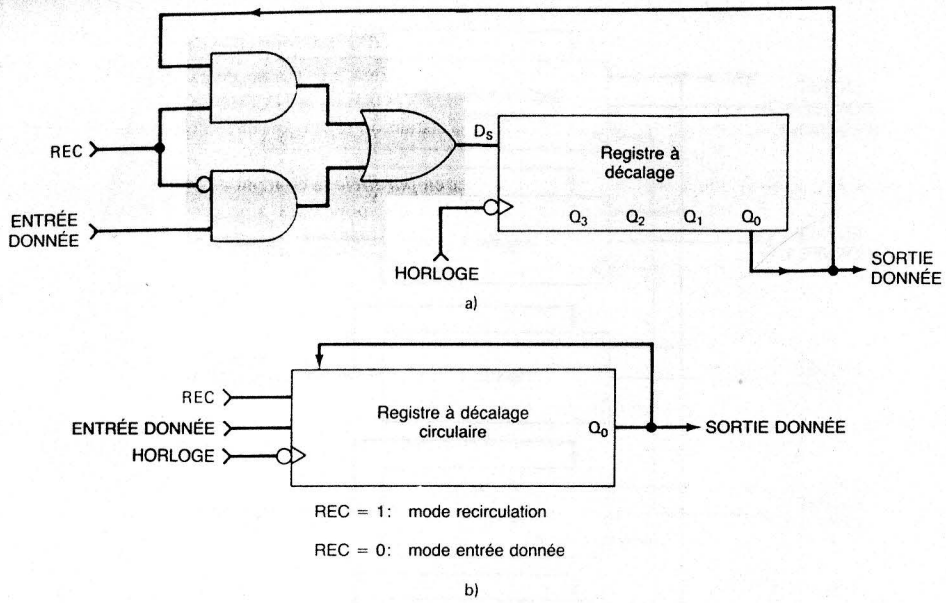


FIGURE 11-41 a) Registre à décalage circulaire; b) Représentation simplifiée.

un registre à décalage; dans la présente figure on en utilise quatre. La donnée arrive au registre par l'entrée série D_S, et elle est reportée dans Q_3, ensuite elle passe de Q_3 à Q_2, de Q_2 à Q_1 et, enfin, de Q_1 à Q_0. La sortie Q_0 est rétrocouplée sur l'entrée série après avoir franchi certains circuits logiques. Ces circuits assurent deux modes de fonctionnement qui sont déterminés par la borne dite de *recirculation REC*. Le niveau sur cette borne fixe la source de la donnée qui atteindra l'entrée série.

MODE RECIRCULATION (*REC* = 1) Dans ce mode, la porte ET du haut est validée et la sortie Q_0 (SORTIE DONNÉE) est appliquée sur D_S. À mesure qu'arrivent les impulsions d'horloge, la donnée dans le registre de décalage recircule conformément à la séquence que voici:

$$Q_3 \rightarrow Q_2 \rightarrow Q_1 \rightarrow Q_0 \rightarrow \textit{SORTIE DONNÉE}$$

Pendant qu'elle recircule dans le registre, la donnée apparaît en outre sur la borne SORTIE DONNÉE, un bit à la fois. Dans ce mode, ENTRÉE DONNÉE est bloquée et n'a aucun effet sur la donnée dans le registre.

MODE ENTRÉE DE DONNÉE (*REC* = 0) Dans ce mode, la porte ET du bas est validée, et le signal sur ENTRÉE DONNÉE est appliqué sur D_S. À mesure qu'arrivent les impulsions d'horloge, la donnée dans le registre est décalée conformément à la séquence que voici:

$$\textit{ENTRÉE DONNÉE} \rightarrow Q_3 \rightarrow Q_2 \rightarrow Q_1 \rightarrow Q_0 \rightarrow \textit{SORTIE DONNÉE}$$

Dans ce cas, il n'y a pas de recirculation de la donnée, puisque la porte ET du haut est bloquée par le zéro sur la borne *REC*. Ce mode est utilisé pour mémoriser, par le biais de ENTRÉE DONNÉE, une nouvelle donnée dans le registre.

La figure 11-41 b) nous fait voir le symbole désignant un registre à décalage circulaire. Les portes de contrôle sont intégrés dans le symbole.

EXEMPLE 11-15

Supposez que la valeur dans le registre de décalage est $Q_3Q_2Q_1Q_0 = 1011$, que *REC* = 1 et que ENTRÉE DONNÉE = 0. Quel sera le niveau sur SORTIE DONNÉE après cinq impulsions d'horloge?

Solution En mode recirculation, ENTRÉE DONNÉE n'a aucun effet. Le tableau suivant illustre le changement de chacune des sorties des bascules à mesure qu'arrivent des impulsions d'horloge.

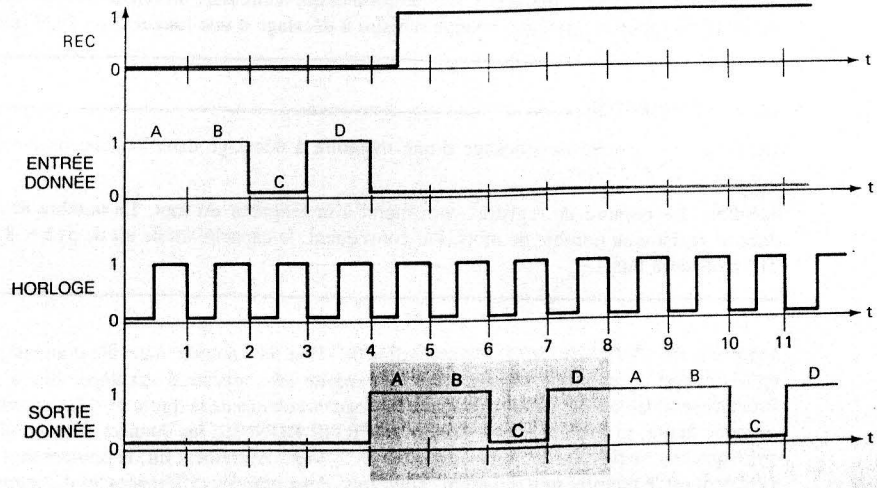
Q_3	Q_2	Q_1	Q_0	
1	0	1	1	Avant la 1re impulsion
1	1	0	1	Après la 1re impulsion
1	1	1	0	Après la 2e impulsion
0	1	1	1	Après la 3e impulsion
1	0	1	1	Après la 4e impulsion
1	1	0	1	Après la 5e impulsion

Donc après la cinquième impulsion Q_0 est à 1 et même chose pour la borne SORTIE DONNÉE.

EXEMPLE 11-16

La figure 11-42 nous montre des formes d'ondes d'entrée et de sortie caractéristiques d'un registre à décalage circulaire. Par hypothèse, le registre ne contient initialement que des zéros. Décrivez la séquence des événements décrite par les formes d'ondes.

FIGURE 11-42 Formes d'ondes d'entrée et de sortie pour les deux modes opératoires d'un registre à décalage circulaire.



Solution Durant les quatre premiers cycles d'horloge, $REC = 0$ et l'appareil se trouve en mode entrée de donnée. Durant cet intervalle, les données aboutissant sur la borne ENTRÉE DONNÉE sont décalées un bit à la fois (A, B, C puis D) aux instants des fronts descendants de HORLOGE. La broche SORTIE DONNÉE demeure à zéro tant que le quatrième front descendant n'a pas envoyé A dans la bascule Q_0. Avant le cinquième FD du signal d'HORLOGE, REC passe au niveau HAUT, soit dans le mode de recirculation. Alors, pour toutes les impulsions ultérieures, la donnée dans le registre recircule, faisant en sorte de répéter sans cesse la séquence des bits A, B, C et D sur la borne SORTIE DONNÉE.

La figure 11-43 est le diagramme de sept registres à décalage circulaires qui sont combinés pour mémoriser une donnée ASCII à 7 bits, transmise continuellement aux circuits d'un écran vidéo. Les entrées *REC* et HORLOGE des registres sont connectées ensemble afin de placer tous les registres dans le même mode et de synchroniser leur décalage. Les sept bornes ENTRÉE DONNÉE portent le code ASCII du mot qui a été mémorisé dans le registre quand $REC = 0$ (mode entrée donnée). Le registre 0 contient le caractère de rang le plus faible de chaque code ASCII, le registre 1, le caractère de rang immédiatement à gauche, et ainsi de suite. Quand les impulsions d'horloge sont appliquées, les sept bits du mot ASCII sont décalés en même temps à l'intérieur des registres; ainsi, les caractères occupent toujours la même position relative dans leur registre respectif. Les sept lignes SORTIE DONNÉE représentent le mot ASCII décalé dans les registres, envoyé à l'écran vidéo pour la régénération de l'affichage, ainsi que rétrocouplé aux entrées des registres.

EXEMPLE 11-17

Un système de visualisation vidéo affiche 25 lignes de 80 colonnes (ou caractères). Pour cet exemple, qu'est-ce qui est changé dans le système de régnération de la figure 11-43?

Solution L'écran reproduit un grand total de $24 \times 80 = 1920$ caractères. Cela signifie qu'on doit avoir 1920 codes ASCII en mémoire et qui recirculent dans la mémoire séquentielle. Donc, pour ce système, chaque registre à décalage a une longueur de 1920 bits.

EXEMPLE 11-18

Quelle est la capacité de stockage d'une mémoire à décalage dans laquelle on emploie 8 registres de 512 bits?

Solution Le nombre de registres correspond à la longueur du mot. Le nombre de bits dans un registre au nombre de mots. Par conséquent, la capacité totale est de 512×8, ou 512 mots de 8 bits.

Mémoire file (PREMIER-ENTRÉ, PREMIER-SORTI) Une file (comme dans file d'attente) est un autre type de mémoire séquentielle qui exploite les registres à décalage. Elle a une structure semblable à celle des registres à décalage circulaires de la figure 11-43, car comme sur cette figure, la file fournit sur la borne SORTIE DONNÉE les données dans le même ordre que leur arrivée sur la borne ENTRÉE DONNÉE. Autrement dit, le premier mot qui a été écrit est le premier mot qui est lu. Toutefois, deux grandes différences les distinguent. D'abord dans les files, la donnée de sortie n'est pas recirculée une fois qu'elle a été fournie;

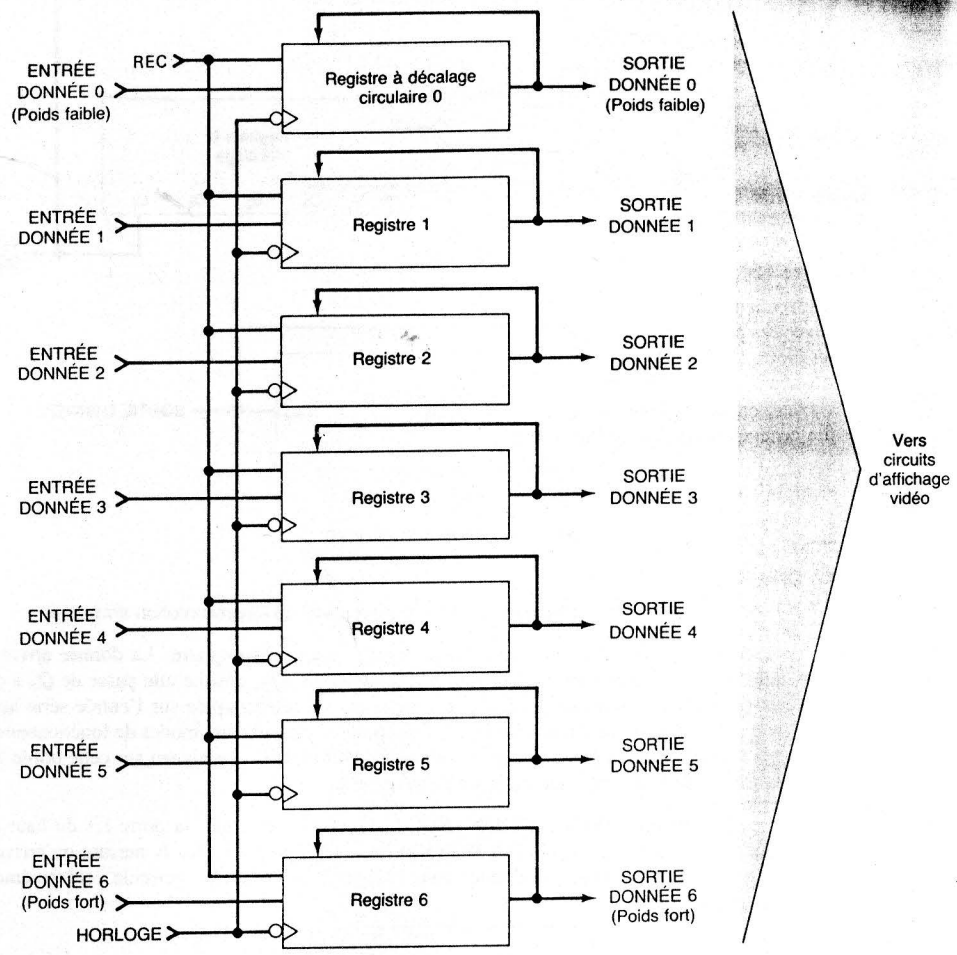


FIGURE 11-43 Assemblage de plusieurs registres circulaires afin de mémoriser et de transmettre sans arrêt des mots de 7 bits.

après leur décalage, les données sont perdues. Deuxième différence: dans une file, le fonctionnement du décalage des données dans les registres est totalement indépendant du fonctionnement de décalage de sortie des données à l'extérieur de la mémoire. En fait, le rythme de décalage des données change selon que l'on mémorise des données ou bien qu'on les décale vers l'extérieur. Cette particularité est notamment pratique pour échanger des données entre des systèmes qui opèrent à des vitesses différentes.

Un exemple courant est le transfert de données entre un ordinateur et une imprimante (figure 11-44). L'ordinateur est en mesure d'envoyer à l'imprimante des données à une cadence beaucoup plus rapide que celle avec laquelle l'imprimante reçoit les données et les imprime. Une file peut jouer le rôle de **tampon de données** entre l'ordinateur et

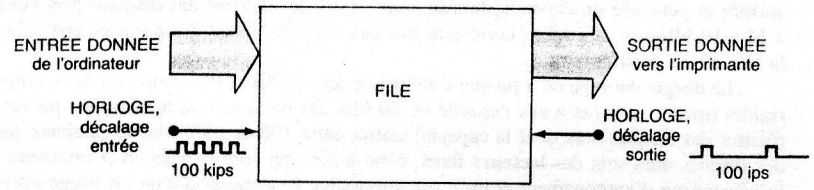


FIGURE 11-44 Une file sert souvent de tampon de données entre un organe rapide, comme un ordinateur, et un organe lent, comme une imprimante.

l'imprimante en acceptant rapidement (par exemple un mot à toutes les 10 μs) les données envoyées par l'ordinateur et en les restituant, plus lentement, à l'imprimante (par exemple un mot à toutes les 10 ms). La file peut également servir de tampon de données dans les situations de transmission de données, lorsqu'un dispositif relativement lent, comme un clavier, envoie des données à un appareil rapide comme un ordinateur. Dans ce cas, la file accepte, à un rythme assez lent, les données du clavier et les mémorise. Par la suite, elle les transmet rapidement à l'ordinateur. De cette façon, l'ordinateur peut continuer à effectuer d'autres tâches pendant que la file de clavier est garnie par les données tapées.

QUESTIONS DE RÉVISION

1. Quels sont les modes opératoires d'un registre à décalage circulaire?
2. Pour le registre à décalage de la figure 11-41, trouvez le niveau sur SORTIE DONNÉE après six impulsions d'HORLOGE si *REC* = 0, ENTRÉE DONNÉE = 1 et les bascules du registre contiennent toutes des zéros au début.
3. Même question que 2, mais cette fois-ci avec *REC* = 1.
4. Décrivez la situation d'une mémoire à décalage qui conserve 64 mots de 9 bits.
5. Qu'est-ce qui distingue une file d'une mémoire à décalage?
6. Dites ce qu'est un tampon de données.

11-20 MÉMOIRES MAGNÉTIQUES

Jusqu'ici, nous n'avons parlé que des mémoires à semiconducteurs, qui sont désormais partout et qui mémorisent les données en conservant une charge ou une tension électrique. Ces dispositifs sont employés en tout premier lieu comme mémoire centrale dans les ordinateurs. Les caractéristiques qui les destinent tout particulièrement à cette tâche sont leur grande vitesse, leur faible encombrement et leur relative simplicité. Le principal inconvénient de ces mémoires est leur volatilité qui les empêche de servir comme mémoires de masse. Dans la section que voici, nous allons vous donner un aperçu des éléments de mémoire dont le principe de stockage est de nature magnétique plutôt qu'électrique. La caractéristique commune à tous ces éléments est leur rémanence (ou non-volatilité).

Tores magnétiques La mémoire à tores magnétiques est un élément de stockage rémanent à accès sélectif et à lecture/écriture. Cette mémoire a été délogée par la mémoire à semiconducteurs, car avant l'arrivée en force de cette dernière, ce sont des tores magnétiques que l'on retrouvait dans la mémoire centrale de la grande majorité des ordinateurs. La cellule élément de la mémoire à tore est un petit anneau toroïdal de matériau ferromagnétique, appelé *tore de ferrite*. Le diamètre type d'un tel tore est 1,25 mm.

Le mécanisme exploité pour conserver des données consiste à enfiler un petit fil dans le centre de l'anneau. Quand un impulsion de courant traverse ce fil, il engendre un flux magnétique dans le tore suivant un sens qui est fonction du sens du courant. À cause du pouvoir de rétention du tore, ce dernier reste aimanté même quand le fil n'est plus parcouru par un courant; en d'autres termes, il est rémanent. Les deux sens d'aimantation possibles figurent la valeur 1 et la valeur 0.

Les données stockées dans les tores sont lues grâce à une aimantation de ces derniers dans le sens 0 (c-à-d. écriture d'un 0) et à un second fil enfilé dans le centre et appelé conducteur de lecture. La valeur de la tension induite dans le conducteur de lecture est supérieure si le tore se trouvait dans l'état 1 plutôt que dans l'état 0. La lecture d'un tore est dite « destructive », puisqu'elle laisse toujours le tore dans l'état 0. Il est donc nécessaire de réécrire la donnée après une opération de lecture.

Les mémoires à tores magnétiques ont des temps d'accès de 100 ns à 500 ns et se retrouvent encore dans certains vieux miniordinateurs et grands ordinateurs. Leur encombrement et leurs circuits d'interface complexes ont fini par les reléguer aux oubliettes.

Mémoires à bulles magnétiques (MBM) La mémoire à bulles magnétiques est une mémoire à semiconducteurs qui conserve une donnée binaire sous l'aspect de minuscules domaines magnétiques (bulles) disséminés sur une mince pellicule de matériau magnétique. La présence ou l'absence d'une bulle à un endroit précis représente un 1 logique ou un 0 logique respectivement. Des champs magnétiques qui varient sans interruption servent à déplacer les bulles dans une boucle à l'intérieur d'un matériau magnétique, un peu comme dans un registre à décalage circulaire. Les données en mouvement passent sous un point de lecture où elles sont communiquées à l'extérieur au rythme de 50 000 bits par seconde. On voit clairement que les MBM sont des mémoires à accès séquentiel.

La rémanence des MBM est leur principal atout: en cas de perte de tension, les données mémorisées ne sont pas perdues, puisque les bulles restent à des endroits fixes. À la reprise de l'alimentation, les bulles recommencent à tourner autour de leurs boucles. Contrairement aux mémoires à semiconducteurs rémanentes, comme les mémoires MORTE, les EPROM et les EEPROM, les MBM conviennent aussi bien à la lecture qu'à l'écriture. De plus, et contrairement aux mémoires sur disques et sur bandes, un système MBM ne renferme aucune pièce mobile et est donc plus silencieux et plus fiable.

Les MBM sont peu encombrantes et dissipent très peu de puissance (environ 1 μW par bit). Une mémoire MBM caractéristique peut mémoriser un million de bits et plus, ce qui est beaucoup plus que n'importe quelle autre mémoire à semiconducteurs existant actuellement. Malheureusement, malgré tous ces avantages les MBM ne peuvent servir de mémoire interne principale dans un ordinateur à cause de leur lenteur: les temps accès typiques sont de l'ordre de 1 à 20 ms. C'est la raison pour laquelle ces mémoires conviennent mieux comme mémoires auxiliaires de masse en remplacement des bandes magnétiques et des unités de disque. À l'heure actuelle, les MBM sont environ 100 fois plus rapides que les disques souples quoiqu'un peu plus coûteuses en raison des circuits de soutien nécessaires. À mesure que les prix baisseront, nous verrons les MBM supplanter progressivement les unités de disque souple plus lentes et moins fiables.

Bande et disque magnétiques Les bandes et les disques magnétiques sont des supports mémoires dont le principe est l'enregistrement ou la lecture de points aimantés sur une surface en mouvement recouverte d'un matériau magnétique. Dans chacun de ces dispositifs, une surface non magnétique lisse est recouverte d'une mince épaisseur d'un matériau magnétique. Par exemple, les bandes magnétiques sont des rubans de plastique sur lesquels on a déposé une couche de matériau magnétique. Par ailleurs, les disques sont des supports plats dont les deux côtés sont recouverts d'un matériau magnétique et qui ressemblent à un disque audio.

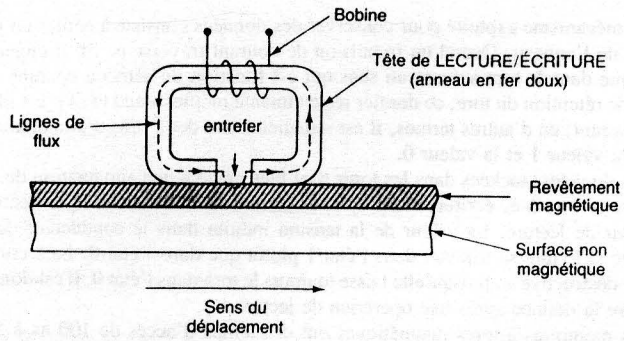


FIGURE 11-45 Principaux éléments permettant d'enregistrer sur une surface magnétique en mouvement.

L'enregistrement et la lecture d'informations binaires se font selon les mêmes principes de base aussi bien pour les bandes que pour les disques. La figure 11-45 illustre le principe fondamental d'enregistrement sur une surface magnétique en mouvement. La tête de lecture/écriture est un anneau de fer doux à haute perméabilité autour duquel on a enroulé une bobine et qui possède un petit entrefer (généralement de 0,0025 cm de largeur). En mode écriture, un courant parcourt la bobine et induit des lignes de flux magnétiques dans l'anneau. Ces lignes de flux sont confinées à l'intérieur de l'anneau jusqu'à l'entrefer qui a une réluctance très élevée au flux magnétique. À cause de cela les lignes de flux sont déviées et parcourent le revêtement magnétique de la surface en mouvement. Par conséquent, des impulsions de courant dans la bobine créent des points d'aimantation sur la surface en mouvement. Ces points demeurent aimantés après le passage de la tête de lecture/écriture.

L'opération de lecture est exactement l'inverse de l'opération d'écriture. Dans ce cas-ci, la bobine sert de ligne de lecture. Quand la surface magnétique passe sous la tête de lecture/écriture, les points qui ont été aimantés produisent un flux à travers l'entrefer qui se retrouve dans l'anneau. Ceci change le flux de l'anneau et induit une tension dans l'enroulement de lecture, tension qui est ensuite interprétée.

La bobine enroulée autour de l'anneau a habituellement une prise centrale, de façon à ce que la moitié de l'enroulement serve à la lecture et l'autre moitié, faite d'un fil plus gros, serve à fournir le courant durant l'opération d'écriture.

Les disques et les bandes sont employés comme mémoires rémanentes lorsqu'il s'agit de conserver en permanence d'imposants volumes de données sur un support que l'on peut effacer et réutiliser. Comme dans ces deux supports l'enregistrement et la lecture passent par un dispositif mécanique, les temps d'accès sont trop lents pour qu'on s'en serve comme mémoire interne d'ordinateur. Leur grande capacité et leur coût par bit très bas les désignent tout spécialement comme mémoires de masse auxiliaires, que l'ordinateur utilise au besoin pour lire des données ou en inscrire.

Disque durs Dans un disque dur, les données sont enregistrées sur les pistes concentriques d'un disque d'aluminium recouverts sur les deux faces d'une couche magnétique. Le diamètre des disques durs s'échelonne entre 3 ½ et 20 pouces; ces disques sont amenés à des vitesses de rotation allant de 1000 à 3600 tr/min au moyen d'un axe d'entraînement. Les disques vendus actuellement ont des temps d'accès compris entre 20 et 100 ms. Il s'agit du temps moyen pris par la tête de lecture/écriture pour se placer au-dessus de la

surface et pour lire ou écrire la donnée. Les débits de transfert des disques durs vont de 1 M à 10 Mbits/s; ces débits correspondent aux taux de lecture des données une fois que la tête est en position.

Le disque dur type de 5 po que l'on trouve dans un IBM PC comprend deux supports rigides (quatre faces) et a une capacité de 20 Mo. De nos jours, la technologie parvient à réaliser des disques durs dont la capacité atteint entre 100 et 1000 Mo. La majeure partie des disques durs sont des **lecteurs fixes**, c'est-à-dire qui sont montés en permanence sur le mécanisme d'entraînement et qu'il est impossible de déposer durant un usage normal. Certaines unités de disque possèdent des disques amovibles, on parle dans leur cas de **chargeurs de disques**.

La grande vitesse de rotation des disques durs donne lieu à un coussin d'air de quelques microcentimètres au-dessus de la surface du disque. Les têtes de lecture/écriture « flottent » sur ce coussin et ne touchent jamais la surface. Cette solution empêche l'usure de la surface du disque et celle de la tête. Si pour une raison quelconque, la tête percute ou rafle la surface du disque, ou si des particules de saletés interfèrent avec le déplacement de la tête, il peut se produire un **écrasement de tête**. Dans un tel incident, la tête fait contact avec la surface du disque et vient rayer ou encaver la surface. Il en résulte une détérioration des données, qui rend généralement inutilisable le disque. Pour réduire au maximum le risque des écrasements de tête, on retrouve dans la plupart des disques durs un mécanisme de nettoyage de la surface qui souffle de l'air filtré. Au moment de la mise hors tension, la tête de lecture/écriture est rangée avant le freinage du disque. À l'inverse, au moment de la mise sous tension, la tête demeure au repos tant que le disque ne tourne pas suffisamment vite pour reconstituer le coussin d'air.

Lecteurs de disquette Une disquette (ou **disque souple**) est un mince disque de Mylar souple inséré dans une pochette de papier. Les deux surfaces du disque sont recouvertes d'un matériau magnétique dans lequel on peut enregistrer les données. On en connaît trois formats: la disquette de 8 po (l'originale), celle de 5 ¼ po (la minidisquette) et celle de 3 ½ po (la microdisquette). Cette dernière disquette est enfermée dans un boîtier de plastique rigide. Une disquette insérée dans un lecteur se met à tourner à une vitesse de rotation comprise entre 300 et 360 tr/min, soit une vitesse nettement plus faible que celle des disques durs.

La capacité de stockage des disquettes varie de 100 ko à 1 Mo, et plusieurs entreprises annoncent pour très prochainement des disquettes 3 ½ po de 4 Mo. Les temps d'accès des disquettes sont généralement 10 fois plus longs que ceux des disques durs, se situant entre 100 et 500 ms. Les débits de transfert des données s'échelonnent entre 250 kbits et 1 Mbits par seconde.

Malgré leur lenteur et leur capacité de stockage plus petite que celle des disques durs, les disquettes possèdent un avantage marqué: on peut facilement transférer les données sur les disquettes d'une machine à une autre ou envoyer des disquettes par la poste.

Bande magnétique Voilà un support informatique qui sert surtout pour la sauvegarde des disques durs. Étant donné que les disques durs conservent fréquemment de volumineuses bases de données (comme le catalogues des stocks, les dossiers des employés, les horaires des étudiants et leur bulletin), un écrasement de tête peut avoir des effets désastreux, à moins que l'on ait fait régulièrement une sauvegarde (recopie) des données sur une bande magnétique. Les atouts les plus remarquables des bandes de sauvegarde sont leur faible coût par bit et leur grande capacité. Une petite cassette, semblable à une cassette audio, peut mémoriser jusqu'à 1000 Mo. Son principal inconvénient est le temps d'accès quand on cherche un bloc de donnée précis (pour s'en convaincre il suffit de penser au temps qu'il faut pour retrouver une chanson sur une cassette). Cependant, pour un support de sauvegarde, un tel inconvénient n'a pas une grande incidence, parce que les données ne

sont pas lues souvent, et quand il faut lire la bande, elle est lue au complet à partir du début.

Mémoire à disque optique Le disque optique numérique (DON) est la dernière en date des technologies de mémoire de masse et son avenir s'annonce très prometteur. Il fonctionne selon le principe de la réflexion ou de la diffusion d'un faisceau laser extrêmement étroit par des renfoncements ou des épaulements microscopiques gravés sur sa surface. L'un des points forts du disque optique est sa capacité: on peut mettre jusqu'à 1000 Mo sur un disque de 5 ¼ po. Parmi ses autres avantages, mentionnons son prix raisonnable et sa tolérance à la poussière. Ses temps d'accès et ses débits de transfert se comparent à ceux des disques durs les plus récents.

Les systèmes à disque optiques existent en trois types dont la classification est fondée sur leur nature inscriptible. Les disques qui sont à lecture seulement sont appelés **disque ROM optique** ou **disque compact (CD-ROM)**. On s'en sert pour stocker de grandes bases de données comme des dictionnaires ou des encyclopédies. Un disque optique qui peut être écrit une seule fois est un **disque WORM (inscriptible une seule fois)**. Les CD-ROM et les WORM ne sont pas effaçables. Le **disque optique à lecture/écriture** peut être lu et écrit autant de fois que l'on veut. À cet égard, il fonctionne comme un disque dur. La surface de ce dernier est différente de celle des autres DON, puisque le matériau magnétique qui la recouvre possède des propriétés magnétiques transformables par un faisceau laser.

La mémoire à disque optique semble être la solution de l'avenir comme support pour les mémoires de masse. Beaucoup d'entreprises mènent d'importants projets de recherche sur de nouvelles techniques de lecture et d'écriture.

QUESTIONS DE RÉVISION

1. Quelle est la principale utilisation des disques et des bandes dans un système informatique?
2. Définissez un disque dur.
3. Qu'est-ce que un écrasement de tête?
4. Donnez les principaux avantages et inconvénients d'une disquette comparativement au disque dur.
5. À quoi sert principalement une bande magnétique?
6. Quels sont les principaux avantages du disque optique numérique?
7. Énumérez les trois types de disque optique?
8. Pourquoi les disques et les bandes ne sont-ils pas utilisées comme mémoire interne dans un ordinateur?

11-21 DÉPANNAGE DES MÉMOIRES VIVES

Tous les ordinateurs contiennent de la mémoire VIVE. En outre, un grand nombre d'ordinateurs généraux et d'ordinateurs spécialisés (comme les contrôleurs à microprocesseur et les ordinateurs de régulation de procédé) contiennent de la mémoire MORTE. Chaque boîtier de mémoire VIVE et de mémoire MORTE qui constitue la mémoire interne d'un ordinateur renferme habituellement des milliers de cellules mémoires. La défaillance d'une seule de ces cellules peut être à l'origine d'une panne de tout le système (dénommé fréquemment « incident système ») ou, à tout le moins, d'une perte de fiabilité. L'essai et le dépannage des mémoires supposent la mise en pratique de techniques que l'on n'exploite pas souvent dans les autres parties d'un système numérique. Parce qu'une mémoire est un assemblage de milliers de circuits identiques agissant comme des emplacements de stockage, tous les tests portant sur son fonctionnement doivent obligatoirement vérifier exactement les cellules qui répondent aux commandes et celles qui ne réagissent pas. Ensuite, il devient possible d'isoler la cause du problème en examinant l'organisation des cellules défectueuses et celles sans défaut. Les problèmes les plus fréquents sont une puce défaillante, un mauvais décodeur de CI, des portes ou des tampons inactifs, ou des connexions défectueuses (des court-circuits ou des chemins de conduction ouverts).

Comme les mémoires VIVES sont à lecture et écriture, leur essai est généralement plus compliqué que celui des mémoires MORTES. Dans la présente section, nous nous proposons d'examiner les procédures habituellement suivies pour tester la mémoire VIVE d'un système et pour interpréter les résultats. L'examen des mémoires MORTES fait l'objet de la section suivante.

Connaissance du fonctionnement La mémoire VIVE illustrée à la figure 11-46 nous servira dans les exemples suivants. Comme nous les répétons depuis le début, la réussite d'une opération de dépannage d'un circuit ou d'un système complexe passe invariablement par une connaissance approfondie du fonctionnement de l'élément. Avant de commencer l'analyse de notre mémoire VIVE, il importe avant tout de comprendre tous les aspects de son fonctionnement.

La capacité totale de notre mémoire est 4K × 8 obtenue en assemblant des modules RAM 1K × 8. Un module est soit une seule puce ou un groupe de puces (par ex. deux puces 1K × 4). Chaque module est raccordé à l'UCT au travers des bus d'adresse et de donnée, et de la ligne de commande $R/\overline{W}$. Ces modules partagent les lignes d'E/S des données. Pendant une lecture, ces lignes sont considérées comme des lignes de sortie que le module sélectionné utilise pour acheminer la donnée réquisitionnée jusque sur le bus de donnée de l'UCT. Durant une écriture, elles agissent comme des lignes d'entrée qui recueillent les données envoyées sur le bus de donnée par l'UCT en vue de leur écriture dans la cellule adressée.

Le décodeur 74LS138 et la porte OU à quatre entrées travaillent de concert pour décoder les six lignes d'adresse de rangs supérieurs afin de générer les signaux de sélection de puce $\overline{K0}$, $\overline{K1}$, $\overline{K2}$ et $\overline{K3}$. Ces signaux valident un module VIVE précis en vue d'une opération de lecture ou d'écriture. L'INVERSEUR est en outre utilisé pour inverser le signal d'horloge de sortie de l'UCT, ϕ_2, afin que le décodeur soit validé seulement quand ce signal ϕ_2 est HAUT. Cette impulsion d'horloge ne survient qu'après une attente suffisante pour permettre la stabilisation des lignes d'adresse après l'imposition de la nouvelle adresse. ϕ_2 se trouve dans le niveau BAS tant que les lignes $R/\overline{W}$ sont en train de changer; cette mesure empêche l'arrivée de pointes de décodage qui pourraient malencontreusement déclencher une puce mémoire et entrer des données non valides.

Chaque module de mémoire VIVE possède ses entrées d'adresse, raccordées aux lignes d'adresse A_0 à A_9 du bus d'adresse de l'UCT. Les lignes d'adresse de rangs supérieurs A_{10} à A_{15} sélectionnent les modules de mémoire VIVE. Le module sélectionné décode les lignes A_0 à A_9 pour localiser la case du mot qui est adressée. Les exemples qui suivent vous montreront comment déterminer l'adresse correspondant à chaque module.

EXEMPLE 11-19

Supposez que l'UCT lance une opération de lecture de ce qui se trouve à l'adresse 06A3 (hex). Déterminez, s'il y a lieu, le module de mémoire VIVE qui est lu.

Solution D'abord, écrivons l'adresse complète en binaire:

A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
0	0	0	0	0	1	1	0	1	0	1	0	0	0	1	1

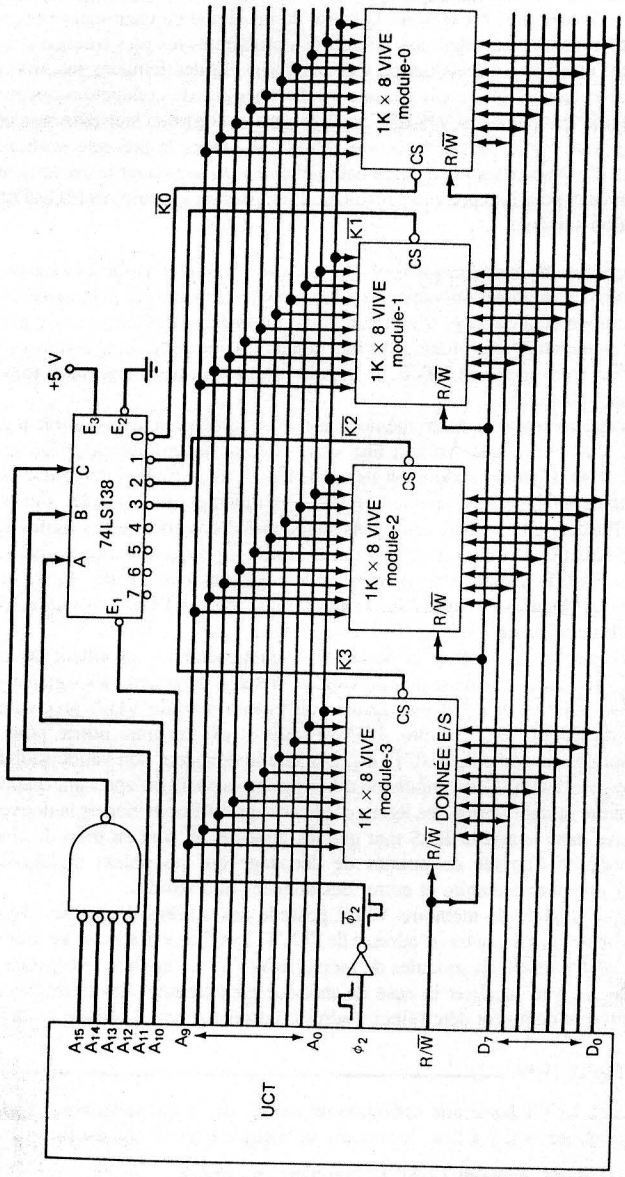


FIGURE 11-46 Mémoire VIVE 4K × 8 raccordée à une UCT.

Vous devriez être en mesure de vérifier que les niveaux A_{15} à A_{10} activent le signal $\overline{K1}$ qui sélectionne le module VIVE-1. Ce module décode ensuite l'adresse sur les lignes A_9 à A_0 afin de localiser la case dont la donnée est placée sur le bus de donnée.

EXEMPLE 11-20

Quel module VIVE sera écrit quand l'UCT lance une opération d'écriture au moyen de l'adresse 1C65?

Solution Le développement en binaire de cette adresse nous permet de constater que $A_{15} = 1$. Ce qui produit un HAUT à la sortie de la porte OU et sur la borne C du décodeur. Comme $A_{11} = A_{10} = 1$, les entrées du décodeur sont désormais 111, ce qui active la sortie 7. Les sorties $\overline{K0}$ et $\overline{K3}$ sont toutes inactives et aucun module VIVE n'est validé. Autrement dit, la donnée placée sur le bus de donnée par l'UCT n'est acceptée par aucun module de mémoire VIVE.

EXEMPLE 11-21

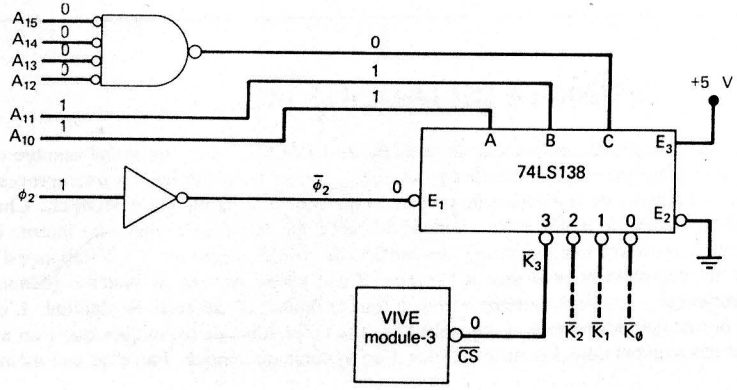
Déterminez le domaine des adresses pour chacun des modules de la figure 11-46.

Solution Chacun des modules emmagasine 1024 mots de 8 bits. Pour déterminer les adresses des mots se trouvant dans un module quelconque, nous devons obtenir les conditions sur le bus d'adresse qui activent les bornes de sélection de cette puce. Par exemple, pour que le module-3 soit sélectionné, il faut que le signal $\overline{K3}$ du décodeur soit au niveau BAS (figure 11-47), ce qui se réalise lorsque $CBA = 011$. Si on procède maintenant à l'inverse pour aboutir aux lignes d'adresse A_{15} à A_{10} de l'UCT, on constate que le module-3 est toujours validé quand l'une des adresses suivantes est sur le bus:

A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0
0	0	0	0	1	1	x	x	x	x	x	x	x	x	x	x

Les x sous les désignations de conducteur A_9 à A_0 signifie « valeur indifférente », parce que les valeurs sur ces lignes n'interviennent pas dans le décodage du module-3. Sur A_9 à A_0 on peut trouver n'importe quelle combinaison entre 0000000000 et 1111111111.

FIGURE 11-47 Exemple 11-21; on peut voir ici les conditions sur le bus d'adresse qui sélectionnent le module VIVE-3.



qui désigne une case du module. Ainsi le domaine complet des adresses pour le module-3 est celui-ci (en remplaçant tous les x par des 0 pour la borne inférieure et par des 1, pour la borne supérieure).

A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	A_8	A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	
0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	$\rightarrow 0C00_{16}$
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	$\rightarrow 0FFF_{16}$

Ce qui donne en définitive 0C00 à 0FFF comme domaine des adresses hexadécimales pour le module-3. Quand l'UCT place une adresse quelconque de ce domaine sur le bus d'adresse, *seul* le module-3 est validé pour la lecture ou l'écriture, selon la valeur de la ligne $R/\overline{W}$.

Une analyse similaire en tous points nous permettrait d'établir les domaines des adresses pour chacun des autres modules. Voici la réponse complète:

Module-0: 0000-03FF
Module-1: 0400-07FF
Module-2: 0800-0BFF
Module-3: 0C00-0FFF

Remarquez que la gamme totale de toutes ces adresses est de 0000 à 0FFF.

Test des circuits logiques de décodage Dans certaines situations, la partie circuits logiques de décodage de la mémoire VIVE (figure 11-47) peut être testée en recourant aux techniques déjà exposées pour les circuits combinatoires. On peut les évaluer en appliquant des signaux aux six lignes d'adresse de poids le plus fort et à ϕ_2 et en observant les sorties du décodeur. Pour parvenir à cela, il faut pouvoir débrancher facilement les lignes d'adresse en provenance de l'UCT. Si l'UCT est une puce microprocesseur dans un support, il suffit simplement de le retirer du support.

Après la déconnexion de l'UCT, on fournit les signaux A_{10}–A_{15} et l'horloge ϕ_2 au moyen d'un circuit de test extérieur en vue de réaliser un essai statique, soit en manipulant des interrupteurs pour créer chaque signal, soit en recourant à un compteur qui parcourt divers codes d'adresse. Une fois les signaux de test appliqués, on peut contrôler les lignes de sortie du décodeur et voir s'ils sortent la bonne réponse. On peut aussi employer les techniques de traçage de signaux pour cerner les causes de défaillance dans les circuits logiques de décodage.

EXEMPLE 11-22

Un test dynamique est effectué sur les circuits de décodage de la figure 11-47; pour cela, on garde $\phi_2 = 1$, et on raccorde les sorties d'un compteur de 6 bits sur les entrées d'adresse A_{10} à A_{15}. Les sorties du décodeur sont surveillées à mesure que le compteur répète son cycle en parcourant tous les codes de 6 bits. Une sonde logique placée sur les sorties du décodeur montre qu'il y a des impulsions sur $\overline{K1}$ et $\overline{K3}$, mais que les sorties $\overline{K0}$ et $\overline{K2}$ restent continuellement au niveau HAUT. Quelles sont les défaillances les plus probables?

Solution Il est possible, quoique hautement improbable, que les deux signaux $\overline{K0}$ et $\overline{K2}$ soient gardés à un niveau HAUT en raison d'un court-circuit interne ou externe à V_{CC}. Une défaillance plus plausible est un conducteur ouvert entre A_{10} et l'entrée A du décodeur, puisque cela aurait l'effet d'un niveau HAUT et interdirait l'activation des sorties paires du décodeur. Il est aussi possible que l'entrée A du décodeur soit court-circuitée à V_{CC}; cependant, cette cause est moins probable, car dans ce cas cela se serait répercuté sur le fonctionnement du compteur qui fournit les adresses.

Test d'un système de mémoire VIVE complet Le test et le dépannage des circuits de décodage ne mettent pas en lumière les problèmes touchant les puces mémoires et leurs connexions avec les bus de l'UCT. Les méthodes les plus courantes de test du fonctionnement *intégral* d'une mémoire VIVE impliquent l'écriture d'agencements connus de 0 et de 1 dans chaque case mémoire et la lecture de cette case afin de constater que les agencements ont été bien reproduits. Bien qu'il existe plusieurs agencements utilisables, le plus employé est celui dit « à damier ». Dans cet agencement, on alterne les 0 et les 1 comme dans 01010101. Une fois que tous les emplacements ont été testés avec ce agencement, on l'intervertit (10101010), et chaque emplacement est de nouveau testé. Remarquez que cette succession de tests détermine l'aptitude de chaque cellule à lire et à mémoriser autant les 1 que les 0. À cause de l'alternance des 0 et des 1, l'agencement en damier décèle aussi toute interaction ou tout court-circuit entre cellules voisines. On exploite d'autres types d'agencement pour tester diverses pannes des puces de mémoire VIVE.

Aucun test de la mémoire ne parvient à dépister toutes les défaillances possibles des mémoires VIVES avec une précision de 100 p. 100, même si on s'assure que chaque cellule stocke et lit des 0 et des 1. Certaines mémoires VIVES fautives dépendent de l'agencement introduit. Par exemple, il se peut fort bien qu'une mémoire VIVE démontre son aptitude à lire et à stocker 01010101 et 10101010, mais qu'elle échoue devant l'agencement 11100011. Même pour une petite mémoire VIVE, le temps nécessaire pour mémoriser et lire toutes les combinaisons possibles dans une cellule est beaucoup trop long. C'est pourquoi on s'accorde pour dire qu'une mémoire VIVE est *probablement* bonne si elle réussit le test du damier, autrement, elle a effectivement un problème.

L'imposition à la main du test du damier (lecture et mémorisation) à des milliers de cellules d'une mémoire VIVE pourra prendre des centaines d'heures, ce qui évidemment est inacceptable. Les tests de combinaison dans les mémoires VIVES sont effectués automatiquement par l'UCT qui exécute un programme de test de la mémoire, ou par un appareil d'essai spécial qui remplace l'UCT et est raccordé sur les bus de la mémoire VIVE. En fait, nombre d'ordinateurs et d'équipements à microprocesseur effectuent automatiquement un test de la mémoire chaque fois qu'ils sont mis sous tension, c'est ce qu'on appelle l'**autotest au démarrage**. Le programme de l'autotest est conservé en mémoire ROM et est exécuté à chaque mise en route du système ou quand l'opérateur l'appelle au clavier. Pendant l'exécution de cet AUTOTEST, l'UCT lit et écrit des agencements de test dans chaque emplacement de la mémoire VIVE et sort des messages à l'intention de l'utilisateur. Ces messages peuvent être simplement l'allumage d'une DEL qui signale une mémoire défectueuse ou bien une description en clair imprimé ou affiché à l'écran. Voici des messages types:

```
test RAM module-3 OK
Toutes les RAM sont correctes
Emplacement 027F défectueux aux positions binaires 6 et 7
```

Devant des messages semblables et à partir de ses connaissances des mémoires VIVES, le dépanneur parvient à savoir comment intervenir pour isoler la défaillance. Voici deux exemples qui illustrent la démarche à suivre pour la mémoire 4K × 8. Afin de vous faciliter la tâche, on a reproduit le schéma de cette mémoire à la figure 11-48. Rappelez-vous (exemple ci-dessus) que les domaines des modules VIVES sont:

Module-0: 0000-03FF
Module-1: 0400-07FF
Module-2: 0800-0BFF
Module-3: 0C00-0FFF

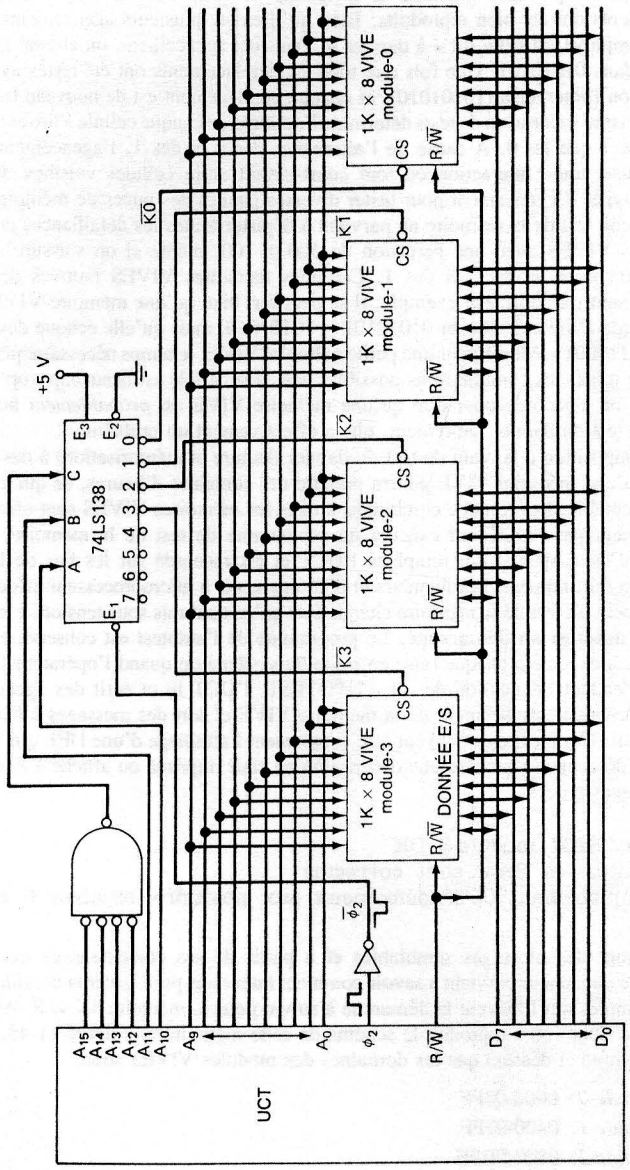


FIGURE 11-48 Mémoire VIVE 4K × 8; reproduction de la figure 11-46.

EXEMPLE 11-23

Voici les messages qui sont imprimés après l'exécution de l'AUTOTEST:

```
test module-0 OK
test module-1 OK
adresse 09A7 défectueuse au bit 8
test module-3 OK
```

Examinez ces messages et déterminez l'intervention suivante.

Solution L'emplacement de l'adresse défectueuse est dans le module-2. Il n'y a aucun circuit extérieur au module-2 qui pourrait donner lieu à une défaillance à cette seule adresse sans en toucher aucune autre. Il faut donc en conclure que la défaillance est interne au module-2. Avant d'en venir à cette conclusion, il faut faire passer l'AUTOTEST à plusieurs reprises afin de s'assurer que les mêmes messages reviennent toujours. Parfois, plutôt qu'une cellule défaillance, du bruit est à l'origine d'un message d'erreur; si c'est le cas, dans les tests successifs il n'y aura pas de message ou bien les messages seront différents. C'est alors le signe qu'il faut rechercher les sources de bruits dans le système, par exemple absence ou mauvais fonctionnement des condensateurs de découplage de l'alimentation, ou bien diaphonie entre lignes de signaux voisins.

EXEMPLE 11-24

Soient les messages suivants sortis par l'AUTOTEST:

```
test module-0 OK
adresse 0400 défectueuse aux bits 0-7
adresse 0401 défectueuse aux bits 0-7
adresse 0402 défectueuse aux bits 0-7
   .      .      .      .
   .      .      .      .
   .      .      .      .
adresse 07FE défectueuse aux bits 0-7
adresse 07FF défectueuse aux bits 0-7
test module-2 OK
test module-3 OK
```

Examinez ces messages, faites une liste des causes possibles et déterminez l'intervention suivante.

Solution De toute évidence, toutes les adresses dans le module-1 sont désignées comme défectueuses, de sorte que le module-1 ne fonctionne pas du tout. Voici les quelques défaillances possibles qui pourraient expliquer ceci:

- une connexion ouverte à l'entrée $R/\overline{W}$ du module-1;
- un chemin ouvert entre $\overline{K1}$ et les bornes sélection de puce du module-1;
- la sortie du décodeur $\overline{K1}$ défectueuse (reste au niveau HAUT);
- le noeud $\overline{K1}/\overline{CS}$ court-circuité à V_{CC} par l'extérieur;
- le module-1 RAM est défectueux (ne réagit pas à $\overline{CS}$);
- la broche V_{CC} ou celle de la masse du module-1 n'est pas branchée.

On peut maintenant employer les techniques courantes de dépannage pour cerner la défaillance.

Comme ces exemples l'illustrent éloquemment, le programme de test de la mémoire est utile pour circonscrire le problème à une zone précise. Le dépanneur met à profit les informations qu'il obtient du test de la mémoire afin de déterminer les défaillances possibles et de localiser l'erreur réelle.

Nous devons souligner en outre qu'il existe certaines défaillances des mémoires VIVES qui empêchent l'UCT d'exécuter le programme d'AUTOTEST au démarrage. Par exemple, une défaillance dans un circuit de mémoire VIVE qui maintient toujours à 0 ou à 1 une ligne de donnée ou d'adresse entraîne un fonctionnement erroné quand l'UCT tente d'exécuter le programme d'AUTOTEST qui se trouve en mémoire MORTE (rappelez-vous que la mémoire MORTE partage les mêmes lignes d'adresse et de donnée que la mémoire VIVE). Dans une telle situation, recherchez les lignes pour lesquelles la tension est constante, ensuite appliquez les techniques habituelles pour localiser la ligne en faute.

QUESTIONS DE RÉVISION

1. Quel est le rôle de ϕ_2 dans le circuit de la mémoire VIVE de la figure 11-46?
2. Qu'est-ce que le test en damier? À quoi sert-il?
3. Définissez l'autotest au démarrage.

11-22 TEST DES MÉMOIRES MORTES

Les circuits de mémoire MORTE dans un ordinateur partagent un grand nombre de points communs avec ceux de la mémoire VIVE (comparez les figures 11-39 et 11-46). La logique de décodage des MORTES peut être testée de la même façon que celle des mémoires VIVES. Les puces de mémoire MORTE, toutefois, exigent d'être testées différemment des puces de mémoire VIVE, principalement parce qu'on ne peut pas y écrire des agencements de 0 et de 1 et essayer ensuite de les lire. Nous disposons de plusieurs méthodes pour vérifier le contenu des puces de mémoire MORTE.

Une solution est d'insérer la puce MORTE dans le support d'un appareil d'essai spécial, habituellement contrôlé par un microprocesseur. Cet appareil peut être programmé pour lire chaque emplacement au moment du test de la mémoire et pour imprimer ce qu'il a relevé. La liste est ensuite comparée avec ce qui normalement doit se trouver dans la mémoire MORTE. Sauf pour les puces de très faible capacité, cette procédure prend un temps énorme.

Dans une autre solution plus efficace, l'appareil contient dans une autre puce MORTE, dite de *référence*, le code sans erreur de ce qui doit être contrôlé. L'appareil est programmé pour lire en parallèle chaque emplacement et sa contrepartie dans la puce de référence et pour les comparer. Évidemment, une telle solution suppose que l'on a à portée de la main la puce MORTE de référence.

Une troisième solution a recours à une **somme de contrôle**. Il s'agit d'un code spécial placé dans la dernière ou les deux dernières cases de la puce MORTE au moment de sa programmation. Ce code est obtenu en additionnant les mots mémorisés dans tous les emplacements de la puce (sauf ceux contenant la somme de contrôle). L'appareil d'essai, pendant qu'il lit chaque donnée de la puce, les additionne afin de constituer sa propre somme de contrôle. Il compare ensuite ce qu'il a obtenu avec ce qui a été mémorisé dans les derniers emplacements de la mémoire MORTE; les deux valeurs doivent être identiques. Si c'est le cas, la probabilité est élevée que la puce MORTE soit bonne, car la probabilité est très faible qu'une combinaison d'erreurs dans la puce se conjugent pour donner la même somme de contrôle. Par contre, si les deux valeurs ne sont pas identiques, la puce de mémoire MORTE a sans conteste un problème.

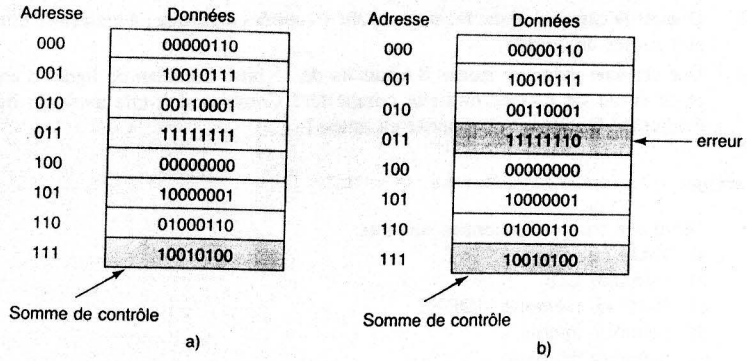


FIGURE 11-49 Méthode de la somme de contrôle appliquée à une mémoire MORTE 8 × 8; a) la puce MORTE est correcte; b) la puce MORTE contient une donnée erronée.

Le principe de la somme de contrôle est illustré à la figure 11-49 a) pour le cas d'une très petite mémoire VIVE. La donnée mémorisée dans la dernière case est la somme sur 8 bits des mots précédents (le report du chiffre de rang le plus fort est éliminé). Au moment de la programmation de cette mémoire MORTE, la somme de contrôle est placée dans la dernière case. La figure 11-49 b) nous montre la somme qui est effectivement lue dans une puce défectueuse dans laquelle on a programmé les données indiquées en a). Remarquez que la donnée erronée est celle dans la case 011. Quand l'appareil d'essai lit les données de la puce erronée, il fait la somme de ce qu'il trouve. À cause d'une erreur, il trouve comme somme 10010011. Quand l'appareil compare la somme qu'il a calculée avec celle mémorisée à l'adresse 111 de la puce, il constate un écart et il désigne cette puce comme défectueuse. Évidemment, on ne sait pas à quel endroit de la puce se trouve l'erreur.

La méthode de la somme de contrôle est exploitée par les ordinateurs et les microprocesseurs pendant un autotest au démarrage afin de vérifier le bon état des mémoires MORTES. Tout comme pour la mémoire VIVE, l'UCT exécute un programme d'autotest de la mémoire MORTE et imprime à la fin un message de la situation. Comme le programme d'autotest est lui-même dans une mémoire MORTE, toute erreur dans cette dernière empêchera la bonne exécution du calcul des sommes de contrôle.

QUESTION DE RÉVISION

1. Qu'est-ce qu'une somme de contrôle? À quoi sert-elle?

PROBLÈMES

SECTIONS 11-1 à 11-3

11-1 La capacité d'une certaine mémoire est 16K × 32. Indiquez le nombre de mots qu'elle peut emmagasiner. Dites le nombre de bits par mot. Dites combien elle contient de cellules mémoires.

11-2 Indiquez le nombre d'adresses différentes dont a besoin la mémoire du problème 11-1.

11-3 Donnez la capacité d'une mémoire ayant 16 entrées d'adresse, 4 entrées de donnée et 4 sorties de donnée.

11-4 Une certaine mémoire stocke 8 kilomots de 16 bits. Combien de lignes d'entrée et de sortie de donnée doit-elle comporter? Combien doit-elle avoir de lignes d'adresse? Quelle est sa capacité en octets?

CONTRÔLE

11-5 Définissez chacun des termes suivants:
a) RAM ou mémoire VIVE
b) mémoire L/E
c) ROM ou mémoire MORTE
d) mémoire interne
e) mémoire de masse

11-6 a) Nommez les trois bus du système de mémoire d'un ordinateur.
b) Quel bus est utilisé par l'UCT pour sélectionner la case mémoire?
c) Quel bus sert à transporter les données entre la mémoire et l'UCT durant une lecture?
d) Quelle est la source des données durant une écriture?

SECTIONS 11-4 et 11-5

11-7 Reportez-vous à la figure 11-6. Déterminez les données de sortie pour chacune des conditions d'entrée que voici:
a) $[A] = 1011$; $CS = 1$
b) $[A] = 0111$; $CS = 0$

11-8 Reportez-vous à la figure 11-7.
a) Quel registre est validé par l'adresse d'entrée 1011?
b) Quel est le code d'adresse qui sélectionne le registre 4?

11-9 Une certaine mémoire MORTE a une capacité de 16K × 4 et une structure interne semblable à celle de la figure 11-7.
a) Combien y a-t-il de registres dans la matrice?
b) Combien y a-t-il de bits par registre?
c) Quel type de décodeur emploie-t-elle?

CONTRÔLE

11-10 a) *Vrai ou faux:* Les mémoires MORTES ne peuvent être effacées.
b) Que signifie la programmation ou la fusion d'une mémoire MORTE?
c) Donnez une définition du temps d'accès d'une mémoire MORTE.
d) De combien d'entrées de donnée, de sorties de donnée et d'entrées d'adresse une mémoire MORTE de 1024 × 4 a-t-elle besoin?

SECTION 11-6

11-11 La figure 11-50 montre comment des données dans une mémoire MORTE peuvent être transférées à un registre externe. Cette mémoire MORTE a les paramètres de synchronisation suivants: $t_{ACC} = 250$ ns et $t_{OE} = 120$ ns. Supposez que la

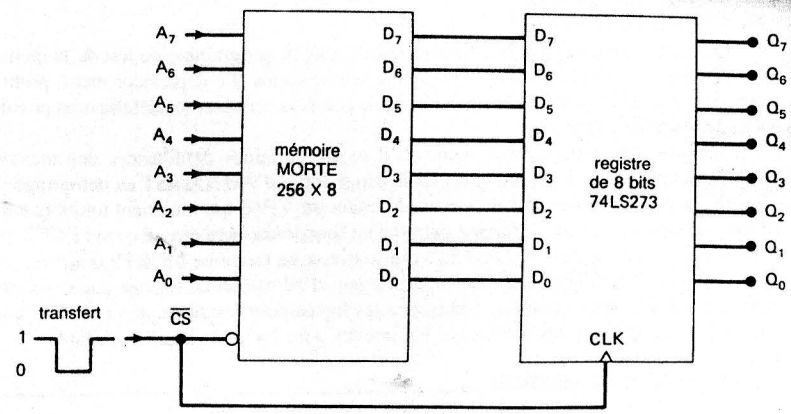


FIGURE 11-50

nouvelle adresse est appliquée à la mémoire MORTE 500 ns avant l'arrivée de l'impulsion TRANSFERT. Déterminez la durée minimale de l'impulsion de transfert qui assure une transmission fiable des données.

11-12 Même problème que 11-11 si l'adresse est changée 70 ns avant l'impulsion TRANSFERT.

SECTION 11-7

11-13 CONTRÔLE

Pour chaque description suivante, dites à quel(s) type(s) de mémoire MORTE elle s'applique.
a) Elle peut être programmée par l'utilisateur, mais jamais effacée.
b) Elle est programmée par le fabricant.
c) Elle peut être effacée et reprogrammée autant de fois que l'on veut.
d) Elle est effaçable par un rayonnement ultraviolet.
e) On peut effacer des mots sélectionnés.
f) Elle utilise des filaments fusibles.
g) Elle peut être complètement effacée en quelques millisecondes.

11-14 Reportez-vous à la figure 11-9 et déterminez les niveaux des sorties de donnée quand $A_1 = A_0 = 1$ et $\overline{EN} = 0$?

11-15 Modifiez la MROM de la figure 11-9 afin que ce qu'on y trouve en mémoire soit $y = 3x + 5$.

11-16 La figure 11-51 nous montre un circuit peu compliqué pour programmer manuellement une EPROM 2732. Chaque broche de donnée de l'EPROM est raccordée à un interrupteur qui peut être réglé à 1 ou à 0. L'adresse est déterminée par un compteur de 12 bits. L'impulsion de programmation de 50 ms est lancée par un monostable chaque fois que l'on appuie sur le bouton poussoir PROGRAMME.
a) Montrez comment ce circuit peut servir à mettre séquentiellement dans les emplacements mémoires de la EPROM les données que l'on veut.
b) Indiquez comment on peut utiliser des 7493 pour constituer le compteur, et un 74121 pour réaliser le monostable.
c) Dites si les rebonds des interrupteurs affectent le fonctionnement du circuit.

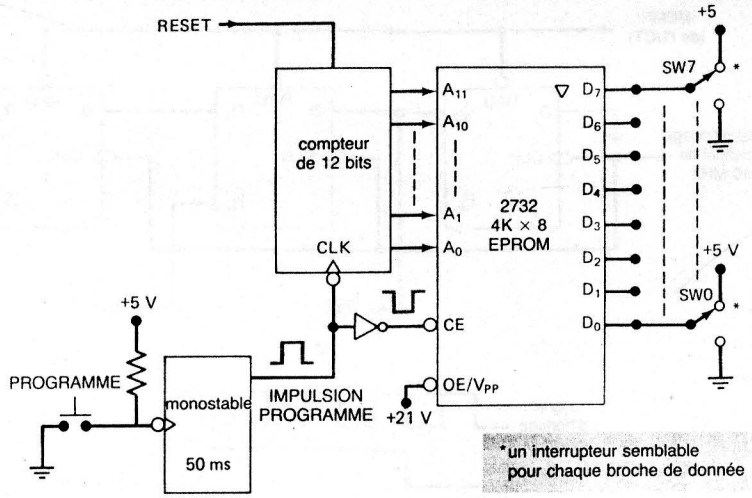


FIGURE 11-51

SECTION 11-8

11-17 Une autre application des mémoires MORTES est la génération de signaux de synchronisation et de commande. La figure 11-52 nous montre une mémoire MORTE 16 × 8 dont les entrées d'adresse sont attaquées par un compteur MODULO-16, ce qui fait que l'adresse qui est appliquée augmente de 1 à chaque impulsion d'entrée. Supposez que la mémoire est programmée comme à la figure 11-6; tracez les formes d'ondes de chacune des sorties de la mémoire MORTE pour un train d'impulsions d'entrée. Ne tenez pas compte des retards de la ROM.

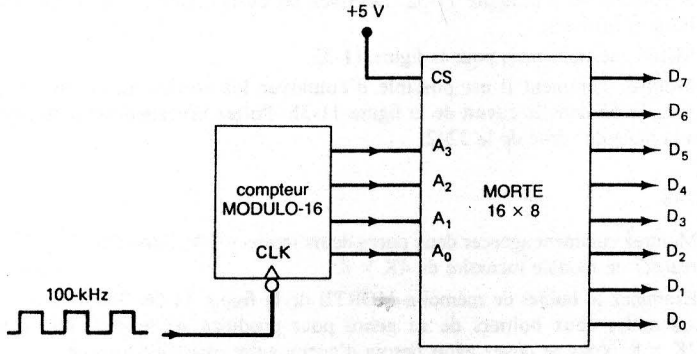


FIGURE 11-52

11-18 Modifiez le programme mémorisé dans la mémoire MORTE du problème 11-17 afin de produire la forme d'onde D_7 illustrée sur la figure 11-53.

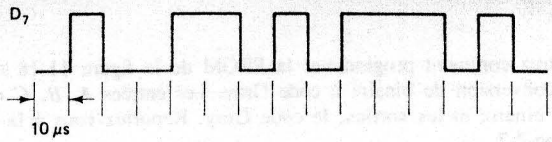
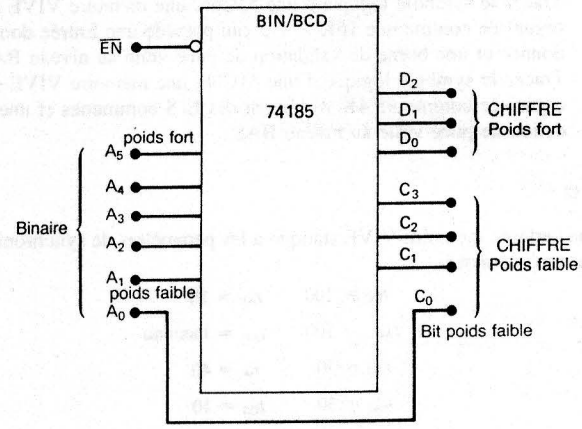


FIGURE 11-53

11-19 La figure 11-54 a) reproduit le symbole logique de la puce 74185, une mémoire MORTE commerciale programmée pour agir comme un *convertisseur binaire-DCB*. Elle convertit une entrée binaire à 6 bits en une sortie DCB de deux chiffres.



a)

Binaire						DCB						
A_5	A_4	A_3	A_2	A_1	A_0	D_2	D_1	D_0	C_3	C_2	C_1	C_0
1	0	0	1	0	1	0	1	1	0	1	1	1
0	0	1	1	1	0	0	0	1	0	1	0	0
1	1	1	1	1	0	1	1	0	0	0	1	0
0	1	1	1	0	0	0	1	0	1	0	0	0
1	1	0	1	0	1							
0	0	0	1	1	0							
						0	0	1	1	0	0	1
						1	0	0	1	0	0	0

b)

FIGURE 11-54

Observez que le bit de poids faible de l'entrée binaire à 6 bits n'est pas connecté à la mémoire MORTE comme une entrée d'adresse; au contraire, il est simplement acheminé à la sortie et devient le bit de poids faible du chiffre de poids faible de la sortie. La figure 11-54 b) présente une partie de la table de vérité pour le contenu de certains emplacements de la mémoire. Examinez cette table et voyez comment s'effectue la conversion binaire-DCB. Ensuite, complétez cette table en inscrivant les valeurs manquantes.

SECTION 11-9

11-20 Montrez comment programmer la PROM de la figure 11-18 a) afin d'obtenir une conversion de binaire à code Gray. Les entrées A, B, C et D figurent le code binaire et les sorties, le code Gray. Reportez-vous à la table 2-4 de la section 2-7.

11-21 Même problème que 11-20, mais en utilisant l'élément PAL de la figure 11-19 a). Rappelez-vous que vous devez exprimer chaque sortie sous la forme d'une addition logique (OU) de quatre termes.

SECTION 11-11

11-22 a) Tracez le symbole logique d'une 51C66, une mémoire VIVE statique CMOS organisée comme une 16K × 1 et qui possède une Entrée donnée, une Sortie donnée et une borne de validation de puce vraie au niveau BAS.

b) Tracez le symbole logique d'une 51C68, une mémoire VIVE statique CMOS organisée comme un 4K × 4 ayant des E/S communes et une borne de validation de puce vraie au niveau BAS.

SECTION 11-12

11-23 Une certaine mémoire VIVE statique a les paramètres de synchronisation suivants (en nanosecondes):

$t_{RC} = 100$	$t_{AS} = 20$
$t_{ACC} = 100$	t_{AH} = inconnu
$t_{CO} = 70$	$t_W = 40$
$t_{OD} = 30$	$t_{DS} = 10$
$t_{WC} = 100$	$t_{DH} = 20$

a) Indiquez le temps qui s'écoule après la stabilisation des lignes d'adresse jusqu'au moment où apparaît une donnée valide sur les sorties durant un cycle de lecture.

b) Dites pendant combien de temps la donnée de sortie demeure valide après que $\overline{CS}$ est retourné au niveau HAUT.

c) Donnez le nombre d'opérations de lecture que l'on peut réaliser par seconde.

d) Indiquez le temps pendant lequel $L/\overline{E}$ et $\overline{CS}$ doivent rester à 1 après la stabilisation de la nouvelle adresse durant un cycle d'écriture.

e) Indiquez le temps minimal durant lequel la donnée d'entrée doit demeurer valide pour qu'une opération d'écriture fiable puisse avoir lieu.

f) Dites pendant combien de temps les entrées d'adresses doivent demeurer stables après que $L/\overline{E}$ et $\overline{CS}$ sont retournés à 1.

g) Indiquez le nombre d'opérations d'écriture qu'on peut effectuer par seconde.

SECTIONS 11-13 à 11-16

11-24 Dessinez le symbole logique d'un TMS4256, une mémoire DRAM 256K × 1. Combien de broches sont économisées sur cette DRAM par le multiplexage d'adresse?

11-25 La figure 11-55 a) schématise un circuit qui génère les signaux $\overline{RAS}$, $\overline{CAS}$ et MUX nécessaires au bon fonctionnement du circuit de la figure 11-19 b). Le signal d'horloge principal de 10 mHz est la base de temps de l'ordinateur. Le signal de

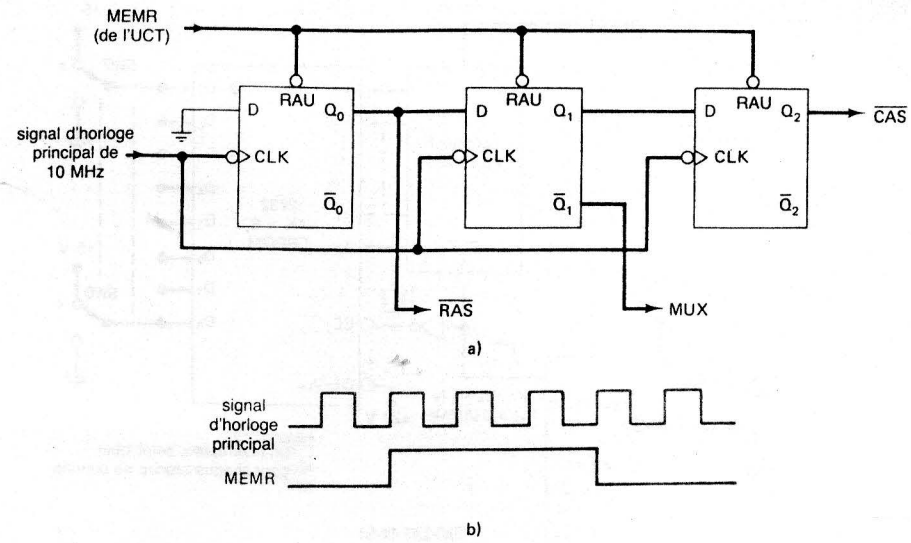


FIGURE 11-55

demande de mémoire (*MEMR*) est produit par l'UCT en synchronisme avec le signal d'horloge principal comme le montre la partie b) de la figure. *MEMR* est normalement à 0 et passe à 1 quand l'UCT veut disposer de la mémoire pour effectuer une opération d'écriture ou de lecture. Déterminez les formes d'ondes en Q_0, $\overline{Q}_1$ et Q_2, puis comparez-les aux formes d'ondes recherchées de la figure 11-31.

11-26 Indiquez comment raccorder deux multiplexeurs 74157 (section 9-7) pour réaliser la fonction de multiplexage dont a besoin le circuit de la figure 11-30 b).

11-27 Reportez-vous à la figure 11-32. Décrivez les événements qui se produisent aux instants marqués.

11-28 Même question, mais pour la figure 11-33.

11-29 Montrez comment il est possible d'employer les sorties du circuit de la figure 11-55 dans le circuit de la figure 11-35. Portez une attention toute spéciale à la table de vérité de la 3242.

SECTION 11-17

11-30 Montrez comment agencer deux puces de mémoire VIVE 2168 (figure 11-23) pour réaliser un module mémoire de 4K × 8.

11-31 Examinez le boîtier de mémoire MORTE de la figure 11-56. Montrez comment assembler deux boîtiers de ce genre pour produire un module mémoire de 2K × 8. Vous ne devez avoir besoin d'aucun autre dispositif logique.

11-32 Modifiez la mémoire de la figure 11-39 afin que sa capacité soit 16K × 8. Utilisez le même type de PROM.

11-33 Modifiez le circuit de décodage de la figure 11-39 pour qu'il soit compatible avec un bus d'adresse de 16 lignes (c.-à-d. addition de A_{13}, A_{14} et A_{15}). Les quatre PROM doivent conserver les mêmes domaines d'adresses.

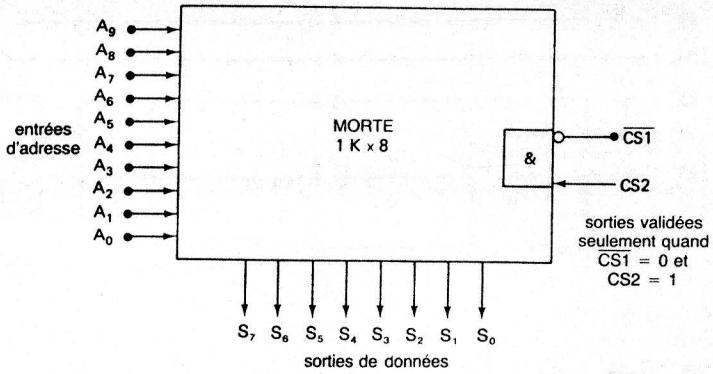


FIGURE 11-56

11-34 Examinez la mémoire de la figure 11-57.

a) Déterminez la capacité totale et la longueur du mot.

b) Notez qu'il y a plus de conducteurs dans le bus d'adresse qu'il en faut pour choisir un emplacement en mémoire. Il ne s'agit pas là d'une situation inhabituelle, particulièrement dans les petits ordinateurs où le nombre de circuits de mémoire est inférieur au nombre maximal de cellules que le bus d'adresse de l'ordinateur peut gérer. Dites quelle mémoire VIVE place une donnée sur le bus de donnée quand $L/\overline{E} = 1$ et que l'adresse est 00010110.

c) Déterminez l'intervalle des adresses pour la combinaison VIVE-0/VIVE-1. Même question pour la combinaison VIVE-2/VIVE-3.

11-35 Tracez le schéma complet d'une mémoire 4K × 4 formée de puces de mémoire VIVE statique ayant les spécifications suivantes: capacité 1K × 1, ligne entrée/sortie commune et deux entrées de sélection de puce vraies au niveau BAS. [*Suggestion:* Il est possible de concevoir ce circuit en utilisant seulement deux inverseurs plus les puces mémoires.]

SECTION 11-19

11-36 Examinez le registre à décalage circulaire de la figure 11-41. La logique de contrôle qui sélectionne la source de la donnée qui est reportée sur l'entrée D_S est en réalité un multiplexeur à deux entrées. Montrez comment il est possible de combiner quatre registres à décalage 74164 (section 7-22) avec un MUX 74157 (section 9-7) pour produire une mémoire registre à décalage 8 × 4.

11-37 Reportez-vous au registre à décalage de la figure 11-43. Chaque registre a 8 bits de longueur. Le contenu de chaque registre est comme suit:

Registre 0: 01111000
Registre 1: 00100001
Registre 2: 01001000
Registre 3: 00000001
Registre 4: 00000100
Registre 5: 10000011
Registre 6: 01111100

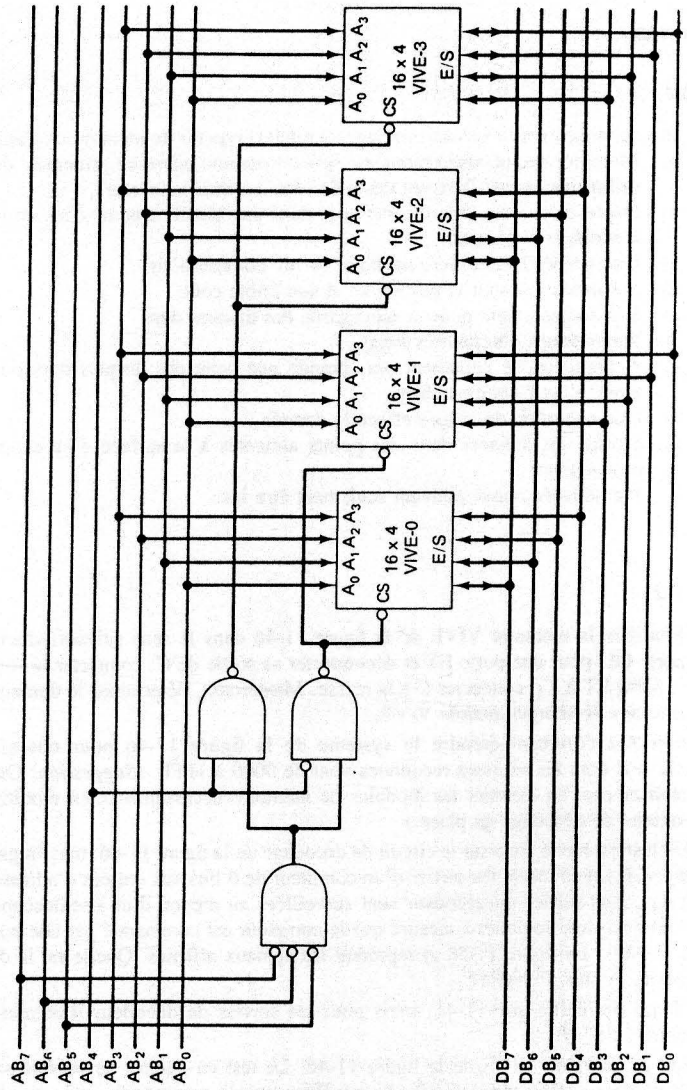


FIGURE 11-57

Quel message en code ASCII est transmis à l'affichage vidéo au moment de l'application des impulsions d'horloge?

SECTION 11-20

CONTRÔLE

11-38 Pour chaque énoncé suivant, indiquez à quel(s) type(s) de mémoire il s'applique.

a) Mémoire magnétique rémanente servant comme mémoire principale dans les ordinateurs avant l'arrivée des mémoires à semiconducteurs.
b) Déplace des domaines magnétiques dans une boucle comme dans un registre à décalage circulaire.
c) Une tête de lecture/écriture flotte sur un coussin d'air.
d) Ses avantages sont la portabilité et son faible coût.
e) Sert fréquemment pour la sauvegarde des disques durs.
f) A des temps d'accès très longs.
g) A une capacité nettement plus grande que celle des disques durs pour des temps d'accès comparables.
h) Une opération de lecture efface la donnée.
i) Stocke des données dans des points aimantés à la surface d'un support en mouvement.
j) Mémoire de masse pouvant seulement être lus.

SECTION 11-21

11-39 Modifiez la mémoire VIVE de la figure 11-46 dans le sens suivant: changer la porte OU pour une porte ET et déconnecter sa sortie de C; connecter la sortie de la porte ET à E_3; connecter C à la masse. Maintenant, déterminez le domaine des adresses de chaque module VIVE.

11-40 Montrez comment étendre le système de la figure 11-46 pour obtenir une 8K × 8 dont les adresses reconnues vont de 0000 à 1FFF. (*Suggestion:* On peut réaliser ceci en ajoutant les modules de mémoire nécessaires et en modifiant la logique de décodage en place.)

11-41 Un test est mené ainsi sur le circuit de décodage de la figure 11-46: tout en gardant $\phi_2 = 1$, on connecte les sorties d'un compteur de 6 bits aux entrées d'adresse A_{10} à A_{13}. Les sorties du décodeur sont surveillées au moyen d'un oscilloscope (ou d'un analyseur logique) à mesure que le compteur est commandé par une horloge de 1 Mhz. La figure 11-58 a) reproduit les signaux affichés. Quelle est la défectuosité la plus probable?

11-42 Même problème que 11-41, mais pour les sorties de décodeur illustrées à la figure 11-58 b).

11-43 Soit la mémoire VIVE de la figure 11-46. Le test en damier ne permet pas de déterminer certains types de défaillance. Par exemple, supposons que la connexion entre l'entrée A et le décodeur soit coupée. L'exécution d'un AUTOTEST en damier sur cette mémoire ne décèle aucune mémoire défectueuse.

a) Expliquez pourquoi cette panne de circuit n'est pas détectée.
b) Comment modifieriez-vous l'AUTOTEST pour parvenir à détecter des défaillances de cette nature?

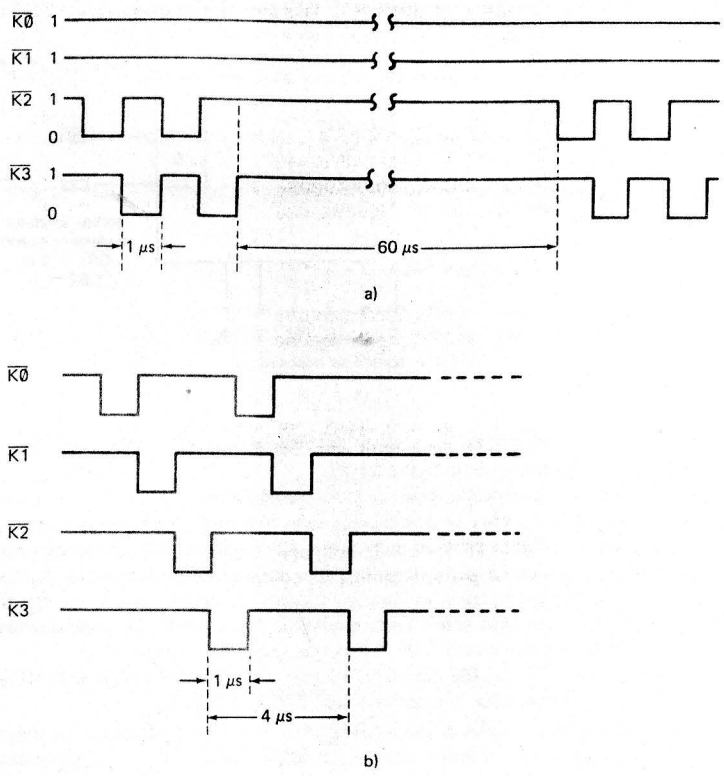


FIGURE 11-58

11-44 Supposez que les modules 1K × 8 de la figure 11-46 sont l'association de deux puces VIVES 1K × 4. Voici les messages qui sont imprimés à la suite de l'exécution de l'autotest au démarrage sur cette mémoire VIVE:

```
test module-0 OK
test module-1 OK
adresse 0800 défectueuse aux bits 4-7
adresse 0801 défectueuse aux bits 4-7
adresse 0802 défectueuse aux bits 4-7
   .    .    .    .
   .    .    .    .
   .    .    .    .
adresse OBFE défectueuse aux bits 4-7
adresse OBFF défectueuse aux bits 4-7
test module-3 OK
```

Examinez ces messages et établissez la liste des défectuosités probables.

11-45 Les messages suivants sont affichés à la suite de l'exécution de l'autotest au démarrage sur la mémoire VIVE de la figure 11-48:

```
test module-0 OK
test module-1 OK
test module-2 OK
adresse 0C00 défectueuse au bit 7
adresse 0C01 défectueuse au bit 7
adresse 0802 défectueuse au bit 7
   .      .      .    .
   .      .      .    .
   .      .      .    .
adresse 0FFE défectueuse au bit 7
adresse 0FFF défectueuse au bit 7
```

Examinez ces messages et établissez la liste des défectuosités probables.

11-46 Quels messages seront affichés après l'autotest de la mémoire VIVE de la figure 11-46 s'il y a un court-circuit entre les sorties $\overline{K2}$ et $\overline{K3}$ du décodeur?

SECTION 11-22

11-47 Soit la mémoire MORTE 16 × 8 de la figure 11-6. Remplacez le mot stocké dans la case mémoire 1111 par une somme de contrôle obtenue à partir des 15 autres mots stockés.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 11-1

1. Voyez texte. **2.** Seize bits par mot; 8192 mots; 131 072 bits ou cellules. **3.** Durant une opération de lecture, un mot est tiré de l'emplacement mémoire et transféré à un autre dispositif; dans une opération d'écriture, un nouveau mot est placé en mémoire et remplace le mot s'y trouvant déjà. **4.** Vrai. **5.** Accès séquentiel: temps d'accès variable qui dépend de l'emplacement physique du mot à traiter; Accès sélectif: temps d'accès constant quel que soit l'emplacement mémoire. **6.** VIVE: mémoire que l'on lit et que l'on écrit en proportions égales; MORTE: mémoire qui est lue beaucoup plus fréquemment qu'elle n'est écrite. **7.** Faux; ses données doivent être rafraîchies fréquemment.

SECTION 11-2

1. 14, 12, 12. **2.** Commande à la mémoire une opération d'écriture ou de lecture. **3.** Dans son état actif, cette entrée valide la mémoire en vue d'une opération de lecture/écriture telle que précisée par l'entrée $L/\overline{E}$; dans son état inactif, cette entrée invalide la mémoire que l'on ne peut alors ni lire ni écrire.

SECTION 11-3

1. Lignes d'adresse, de donnée et de commande. **2. et 3.** Voyez texte.

SECTION 11-4

1. Vrai. **2.** Application des entrées d'adresse nécessaires; activation de l'entrée de commande; apparition de la donnée sur les sorties de donnée.

SECTION 11-5

1. $A_3A_2A_1A_0 = 1001$. **2.** Le décodeur de sélection de la rangée active l'une des sorties de validation de tous les registres de la rangée sélectionnée; le décodeur de sélection de la colonne active l'une des entrées de validation de tous les registres de la colonne sélectionnée; les tampons de sorties transfèrent la donnée du bus de donnée interne aux broches de sortie de la mémoire MORTE quand l'entrée *CS* est activée.

SECTION 11-7

1. Faux; par le fabricant. **2.** Une PROM peut être programmée une fois par l'utilisateur; elle ne peut pas être effacée ni reprogrammée. **3.** Vrai. **4.** Par exposition à des rayonnements ultraviolets. **5.** Vrai. **6.** Voyez texte. **7.** Une EEPROM peut être programmée sans qu'on doive la retirer du circuit; on peut effacer des mots choisis; le temps d'effaçage n'est que de 10 ms; l'impulsion de programmation est plus courte que celle nécessaire pour une EPROM.

SECTION 11-8

1. Les programmes d'un micro-ordinateur stockés dans une mémoire MORTE. **2.** À la mise sous tension, l'ordinateur exécute un petit programme d'amorçage se trouvant en MORTE et qui sert à initialiser les ressources matérielles et à charger le système d'exploitation conservée dans une mémoire de masse (disque). **3.** Le circuit qui prend en entrée un type de code et le convertit en un autre type de code. **4.** Les configurations des points qui correspondent aux caractères. **5.** Compteur, mémoire MORTE, CNA, filtre passe-bas.

SECTION 11-9

1. Un CI qui contient un grand nombre de portes, de bascules et de registres dont les connexions sont modifiables par l'utilisateur pour réaliser des fonctions spécifiques. **2.** $O_1 = A$. **3.** Un filament fusible intact. **4.** Une connexion câblée. **6.** L'élément PAL renferme une matrice ET programmable; la PROM a une matrice OU programmable. **8.** Un élément FLPA renferme à la fois une matrice ET et une matrice OU programmables. **9.** Il faut fondre le fusible et tester le fonctionnement d'un élément PLD afin d'établir qu'il est conforme aux informations contenues dans sa mémoire.

SECTION 11-11

1. L'adresse voulue est appliquée sur les entrées d'adresse; $L/\overline{E} = 1$, *CS* ou *CE* est activé.
2. Pour réduire le nombre de broches.
3. Vingt, y compris V_{CC} et la masse.

SECTION 11-12

1. Les cellules SRAM sont des bascules; les cellules DRAM sont des condensateurs.
2. CMOS. **3.** Mémoire VIVE. **4.** UCT.
5. Durées des cycles de lecture et d'écriture.
6. Faux, quand $\overline{WE}$ est BAS, les broches I/O se comportent comme des entrées de donnée, quel que soit l'état de $\overline{OE}$ (seconde entrée du tableau des modes).

SECTION 11-13

1. Généralement plus lente, et a besoin d'être rafraîchie. **2.** Faible consommation, grande capacité; coût du bit plus économique.
3. IRAM; les contrôleurs de mémoire VIVE dynamique.

SECTION 11-14

1. 256 rangées et 256 colonnes. **2.** Cela réduit le nombre de broches sur la puce. **3.** 1M = 1024K = 1024 × 1024. Donc, il y a 1024 rangées par 1024 colonnes. Comme $1024 = 2^{10}$, cette puce a besoin de 10 entrées d'adresse.
4. $\overline{RAS}$ sert à mémoriser l'adresse de la rangée dans le registre de l'adresse de la rangée de la DRAM; $\overline{CAS}$ sert à mémoriser l'adresse de la colonne dans le registre d'adresse de la colonne.
5. *MUX* multiplexe l'adresse intégrale entre l'adresse de la rangée et l'adresse de la colonne, qui deviennent les entrées de la DRAM.

SECTION 11-15

1. Voyez texte. **2.** Voyez texte.

SECTION 11-16

1. Il fournit l'adresse des rangées à la DRAM pendant les cycles de rafraîchissement. **2.** Le multiplexage de l'adresse et l'opération de rafraîchissement.

SECTION 11-17

1. Seize. **2.** Quatre. **3.** Faux, quand on étend la capacité d'une mémoire, chaque puce est sélectionnée par une sortie différente du décodeur (voyez figure 11-39). **4.** Vrai.

SECTION 11-18

1. Pile de secours pour les RAM CMOS et les NVRAM. **2.** Comme les SRAM et les EEPROM ont leurs cellules dédoublées, la puce renferme 2 × 512 × 8 = 8192 cellules.

SECTION 11-19

1. Mode recirculation; mode entrée de donnée.
2. HAUT. **3.** BAS. **4.** Elle est formée de neuf registres à décalage circulaires connectés comme à la figure 11-43. **5.** Les données peuvent être décalées en entrée et décalées en sortie à des rythmes différents. **6.** Une FILE sert à transférer des données entre des dispositifs ayant des débits de transfert différents.

SECTION 11-20

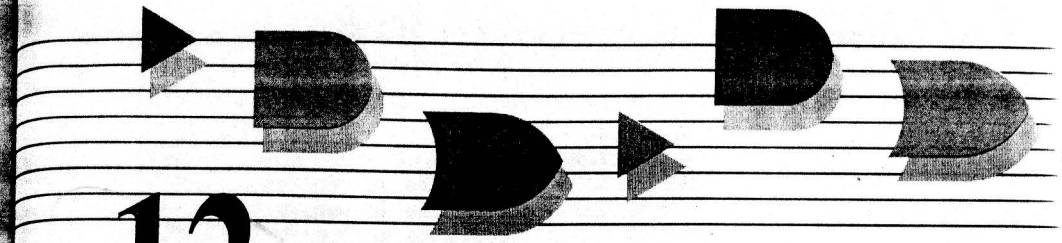
1. Mémoire de masse auxiliaire. **2.** Un disque dur dont le disque ne peut pas être retiré de l'axe d'entraînement. **3.** Le contact entre la tête de lecture/écriture et le disque qui endommage la surface. **4.** Avantages: bon marché, portable; inconvénients: lent, faible capacité. **5.** Support de sauvegarde pour les disques durs. **6.** Grande capacité, faible coût, tolérance à la poussière. **7.** Disque optique ROM, WORM, disque optique à lecture/écriture. **8.** Leurs temps d'accès sont trop lents.

SECTION 11-21

1. Empêche les surtensions de décodage par la désactivation du décodeur au moment du changement des lignes d'adresse. **2.** Une façon de tester des mémoires VIVES est d'écrire un agencement en damier (d'abord 01010101, puis 10101010) dans chaque emplacement mémoire et de les lire. On détecte ainsi les courts-circuits et interactions entre cellules adjacentes. **3.** Un test automatique de la mémoire VIVE effectué au démarrage de l'ordinateur.

SECTION 11-22

1. C'est un code placé dans le dernier ou les deux derniers emplacements de la mémoire MORTE, et qui correspond à la somme des données mémorisées dans tous les emplacements de la mémoire. On s'en sert comme valeur pour déterminer s'il y a des erreurs dans une mémoire MORTE.

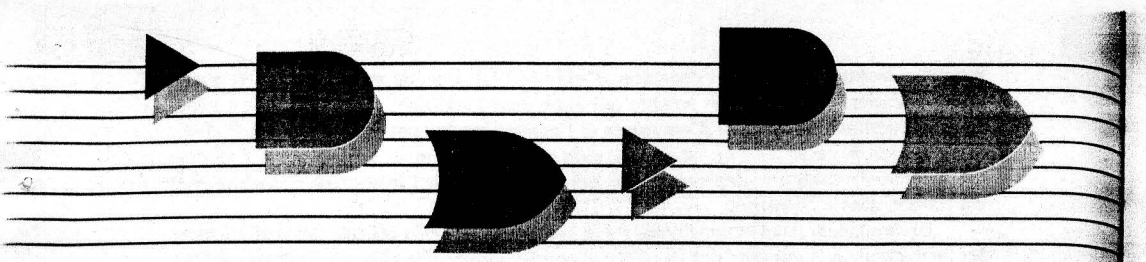


12

INTRODUCTION AUX MICROPROCESSEURS ET AUX MICRO-ORDINATEURS

Contenu

12-1 Définition d'un ordinateur
12-2 Le raisonnement dans un ordinateur
12-3 Agent secret 89
12-4 Organisation élémentaire d'un système informatique
12-5 Éléments de base d'un micro-ordinateur
12-6 Mots machines
12-7 Mots d'instruction
12-8 Exécution d'un programme en langage machine
12-9 Structure type d'un micro-ordinateur
12-10 Opérations de lecture et d'écriture
12-11 Conclusion



Objectifs

À la fin de ce chapitre, vous serez en mesure de:

- Décrire le fonctionnement et le rôle de chacun des cinq éléments de base de la structure d'un ordinateur.
- Évaluer les avantages et les inconvénients des divers formats de mots machines.
- Comprendre la différence entre un microprocesseur et un micro-ordinateur.
- Analyser les cycles de prise en charge et d'exécution durant le déroulement d'un programme informatique.
- Bien saisir le rôle fonctionnel des différents types de bus et de leurs signaux dans un micro-ordinateur.
- Décrire la séquence des événements qui se produisent durant les cycles de lecture et d'écriture pendant l'exécution d'un programme.

Introduction

C'est maintenant un lieu commun que d'affirmer que les microprocesseurs et les micro-ordinateurs ont révolutionné l'industrie de l'électronique et qu'ils ont désormais une incidence importante sur tous les aspects de notre vie. Les techniques d'intégration à grande échelle ont permis de réduire à un tel point la taille des ordinateurs, qu'il ne se passe pas un jour sans qu'un concepteur décide d'en implanter un dans une nouvelle application.

Le présent chapitre est consacré à l'étude des principes fondamentaux des micro-ordinateurs. Comme il existe des livres qui ne traitent que des ordinateurs, le but visé dans celui-ci est de développer une assise solide pour des études ultérieures. Tous les principes que nous allons introduire, même si on ne parle que de micro-ordinateurs, sont valables pour tous les types d'ordinateur, quelle que soit leur taille.

12-1 DÉFINITION D'UN ORDINATEUR

Un ordinateur numérique est une combinaison d'éléments numériques et de circuits qui se concertent pour exécuter une séquence *programmée* d'opérations nécessitant le minimum d'interventions humaines. Cette séquence est appelée un **programme**. Un tel programme est un ensemble d'instructions codées conservées dans la mémoire interne de l'ordinateur avec les données sur lesquelles ce programme travaille. Quand un ordinateur reçoit l'ordre d'exécuter un programme, il suit ces instructions en respectant l'ordre de leur mémorisation et cela, jusqu'à ce que le programme prenne fin. Toutes ces actions sont effectuées à une très grande vitesse et sans erreur.

12-2 LE RAISONNEMENT DANS UN ORDINATEUR

Un ordinateur ne raisonne pas! Pour la simple raison qu'un *programmeur* y a installé un *programme* d'instructions et des données qui stipulent à la machine, et dans les moindres détails, toutes les étapes qu'elle doit suivre et à quel moment elle doit les exécuter. Un ordinateur n'est rien d'autre qu'une machine ultra rapide qui traite des données, résout des problèmes, prend des décisions, tout cela sous la gouverne d'un programme. Si le programmeur s'est trompé dans le programme ou dans les données, l'ordinateur fournira des données erronées. D'ailleurs, on a coutume de dire en informatique: « Ce qui sort ne peut être meilleur que ce qui entre ».

La question à se poser est sans doute: comment un ordinateur s'y prend-il pour exécuter un programme? Habituellement, on répond à cette question en examinant le schéma de l'architecture d'un ordinateur (organisation de ses divers composants) et en suivant pas à pas le travail accompli par un ordinateur pendant son traitement d'un programme. Il s'agit là d'une démarche que nous allons suivre un peu plus tard. Pour le moment, utilisons une analogie farfelue qui, malgré son invraisemblance, met en évidence plusieurs des notions mises en oeuvre dans le fonctionnement d'un ordinateur.

12-3 AGENT SECRET 89

L'agent secret 89 se voit confier la mission de trouver dans combien de jours un chef d'état important doit être assassiné. Son contact l'informe que cette information se trouve dans une série de casiers postaux. Pour empêcher quiconque de percer ce secret, on a dispersé l'information dans dix casiers différents. Son contact lui remet 10 clés et les consignes suivantes:

1. l'information cachée dans chaque casier est codée;
2. ouvrez le casier 1 en premier, puis suivez les directives que vous y trouverez;
3. ensuite, faites de même pour tous les autres casiers dans l'ordre, sauf indication contraire dans les consignes;
4. l'un des casiers est piégé, si vous l'ouvrez il explose!

L'agent 89 muni de ses 10 clés et de son guide de décodage se rend au bureau de poste. La figure 12-1 nous montre le contenu décodé de chacun des 10 casiers postaux. Supposons que vous soyez l'agent 89; votre premier geste est d'ouvrir le casier 1 et d'effectuer l'instruction qu'on vous demande de faire dans l'espoir de savoir le nombre de jours avant l'acte fatidique. Évidemment, ce travail est plus facile pour vous que pour l'agent 89 qui doit au préalable décoder les instructions. La réponse est donnée au paragraphe suivant.

Si vous avez suivi correctement les instructions, vous devez avoir comme réponse, en arrivant dans le casier 6, la valeur 17. Si vous avez commis une erreur, vous avez sans doute ouvert le casier 7, ce qui a provoqué votre fin tragique. Pendant votre périple d'un

① Ajouter le numéro stocké dans le casier ⑨ à votre numéro de code d'agent.	② Diviser le résultat précédent par le nombre stocké dans le casier ⑩.
③ Soustraire le numéro mémorisé dans le casier ⑧.	④ Si le résultat du dernier casier n'est pas égal à 30, aller au casier ⑦ ; autrement passer au casier suivant.
⑤ Soustraire 13 de la dernière réponse.	⑥ FIN. Vous avez votre réponse.
⑦ BOOM désolé!	⑧ 20
⑨ 11	⑩ 2

FIGURE 12-1 Dix casiers postaux et leur message codé pour l'agent 89.

casier à l'autre, vous avez essentiellement procédé comme un ordinateur. Nous allons analyser les opérations et les notions mises en cause dans notre analogie de l'agent secret et montrer la relation avec un ordinateur réel.

Pour ceux qui n'ont pas fait le rapprochement, les casiers représentent la **mémoire** de l'ordinateur où sont conservées les **instructions** et les **données**. Les casiers postaux 1 à 6 renferment des instructions que l'agent secret doit suivre (ou exécuter), tandis que les casiers 8, 9 et 10 contiennent des données utilisées par les instructions. (Le contenu du casier 7, à notre connaissance, n'a pas d'équivalent dans le monde des ordinateurs.) Les numéros repérant chaque casier correspondent aux **adresses** des emplacements en mémoire.

Trois catégories d'instructions différentes sont identifiables dans les casiers 1 à 6. Les casiers 1, 2, 3 et 5 correspondent à des instructions *arithmétiques*. Le casier 4 correspond à une instruction de *prise de décision* appelée *saut conditionnel* ou *branchement conditionnel*. Cette instruction demande à l'agent (ou ordinateur) de décider s'il doit aller à l'adresse 7 ou bien continuer dans l'ordre et se rendre en 5. C'est le résultat des opérations arithmétiques précédentes qui dicte la décision à prendre. Le casier 6 renferme une instruction de contrôle qui n'a pas besoin de donnée ni de renvoi à une adresse (numéro de casier). Cette instruction *FIN* indique à l'agent que la procédure est terminée (le programme est achevé) et qu'on arrête ici.

Chacune des instructions, autant arithmétique que de branchement, est formée de deux parties — une **opération** et une **adresse**. Par exemple, la première partie de la

première instruction précise l'opération d'addition. La seconde partie indique à quelle adresse (casier 9) on trouve le nombre qui entre dans l'addition. Cette donnée est généralement désignée par le nom **d'opérande** et l'adresse par le nom **d'adresse d'opérande**. L'instruction du casier 5 est un cas spécial; dans cette instruction, on ne trouve pas d'adresse d'opérande. Au contraire, l'opérande (donnée) qui fait partie de l'instruction de soustraction est fournie dans l'instruction.

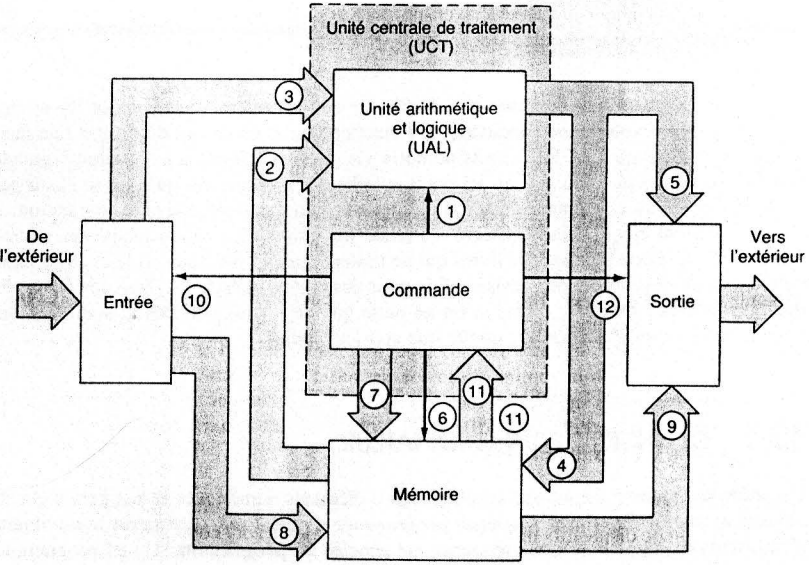
Un ordinateur, tout comme notre agent secret, décode puis exécute *séquentiellement* les instructions mémorisées. Les instructions sont exécutées à la suite en commençant par la première; la séquence est respectée tant qu'une instruction de *branchement* (comme dans le casier 4) ne force pas le programme à aller vers une instruction autre que la suivante. Quand un branchement a eu lieu, le programme poursuit son exécution séquentiellement en partant de la nouvelle adresse fournie.

Voilà tout ce que l'on peut dégager de l'analogie de l'agent secret. Chacune des notions que l'on vient d'esquisser sera détaillée dans les sections suivantes. Heureusement, cette analogie vient décrire un principe de fonctionnement dont vous apprécierez toute l'importance lorsque nous traiterons les ordinateurs de façon plus technique.

12-4 ORGANISATION ÉLÉMENTAIRE D'UN SYSTÈME INFORMATIQUE

Tous les ordinateurs contiennent cinq éléments ou composants incontournables: *l'unité arithmétique et logique* (UAL), *la mémoire*, *l'unité de commande*, *l'unité d'entrée* et *l'unité de sortie*. La figure 12-2 illustre les connexions de base entre chacune de ces composantes. Les flèches dans ce diagramme indiquent le sens de la circulation des données ou des commandes. Vous pouvez voir deux types de flèches: les flèches larges correspondent

FIGURE 12-2 L'organisation élémentaire d'un ordinateur.



aux transferts de données ou d'informations sur un faisceau de plusieurs conducteurs parallèles, et les flèches étroites, aux transferts des signaux de commande, représentent normalement quelques conducteurs seulement. Chacune de ces flèches est également désignée par un numéro, auquel nous renvoyons dans la description suivantes.

Unité arithmétique et logique L'UAL est la zone de l'ordinateur où sont effectuées les opérations arithmétiques et logiques sur les données. La nature de l'opération à mener sur les données est déterminée par les signaux de l'unité de commande (flèche 1). Les données qui sont traitées proviennent soit de la mémoire (flèche 2), soit de l'unité d'entrée (flèche 3). Les résultats du traitement par l'UAL sont transférés dans la mémoire (flèche 4) ou à l'unité de sortie (flèche 5).

Unité de mémoire La mémoire a pour rôle de conserver des groupes de chiffres binaires (mots) qui sont soit des instructions (formant un programme) que l'ordinateur est tenu d'exécuter, soit des données dont le programme a besoin. La mémoire agit également comme unité de stockage temporaire pour accueillir les résultats des opérations arithmétiques ou logiques (flèche 4). L'unité de commande a la main haute sur la mémoire (flèche 6) et ce sont les signaux qui proviennent de cette unité qui imposent une lecture ou une écriture de la mémoire. Un certain emplacement en mémoire est utilisé par l'unité de commande quand cette dernière précise son adresse (flèche 7). Dans la mémoire et toujours sous la régie de l'unité de commande, on peut inscrire des données fournies par l'UAL ou par l'unité d'entrée. Également, on peut lire le contenu de la mémoire et envoyer ces renseignements à l'UAL ou à l'unité de sortie.

Unité d'entrée Cette unité est constituée de tous les dispositifs servant à prélever des informations et des données de l'extérieur de l'ordinateur et à les transférer dans la mémoire (flèche 8). L'unité de commande détermine à quel endroit les données entrées sont acheminées (flèche 10). C'est par l'unité d'entrée que l'on introduit le programme et les données dans la machine avant sa mise en route. C'est aussi par cette unité que l'on passe pour introduire des données dans l'UAL avec un dispositif extérieur comme un clavier, des interrupteurs, des téléscripteurs, des lecteurs de cartes perforées, des unités de disques magnétiques, des unités de bande magnétique, des convertisseurs analogiques-numériques (CAN).

Unité de sortie L'unité de sortie regroupe les éléments que l'on a prévu pour transférer les données et les informations de l'ordinateur vers le « monde extérieur ». Ces dispositifs de sortie reçoivent leurs ordres de l'unité de commande qui envoie des données gardées en mémoire ou des résultats calculés par l'UAL. Les données sont par la suite traduites dans une forme utilisable à l'extérieur. Parmi les exemples d'éléments de sortie, mentionnons: les afficheurs DEL, les voyants indicateurs, les imprimantes, les unités de disque et de bande, les écrans de visualisation et les convertisseurs numériques-analogiques (CNA).

Pendant que l'ordinateur exécute le programme, il est habituel qu'il doive soumettre au monde extérieur des signaux de commande ou des données. Par exemple, un système informatique pourra avoir comme dispositif de sortie une imprimante par ligne. Alors, l'ordinateur enverra des signaux en vue d'imprimer les résultats sur papier. Dans les micro-ordinateurs de plus petite taille, les résultats obtenus sont communiqués au moyen de voyants ou d'afficheurs à DEL.

Interfaçage L'aspect le plus important des unités E/S concerne l'*interfaçage*, c'est-à-dire la liaison de dispositifs dissemblables de façon à les rendre compatibles et en mesure de travailler ensemble parfaitement. *L'interfaçage informatique* est définie plus particulièrement comme la synchronisation de la transmission numérique entre un ordinateur et ses organes d'entrée/sortie extérieurs.

Nombre d'organes d'entrée/sortie ne sont pas immédiatement compatibles avec l'ordinateur, pour des raisons aussi diverses que la vitesse de fonctionnement, le format des données (hexadécimal, ASCII, binaire), le mode de transmission des données (série, parallèle) et le niveau du signal logique. C'est pour de telles raisons que l'on associe à ces dispositifs des circuits d'interface spéciaux qui leur permettent de communiquer avec la mémoire, l'unité de commande et l'UAL de l'ordinateur. Un exemple courant est l'écran de visualisation ou moniteur qui agit autant comme organe d'entrée qu'organe de sortie. Le moniteur transmet et reçoit des données de façon sérielle (un bit à la fois), alors que l'ordinateur manipule les données de façon parallèle. Donc, un écran a besoin d'un circuit qui transforme les données séries en données parallèles comprises par l'ordinateur.

Unité de commande La fonction de l'unité de commande doit désormais vous paraître évidente, c'est-à-dire diriger les opérations de toutes les unités en fournissant les signaux de synchronisation et de commande. Dans un sens, c'est un peu le chef d'orchestre qui s'assure que tous les joueurs du système « suivent la mesure ». Cette unité contient les circuits logiques et de synchronisation qui produisent les signaux nécessaires à l'exécution de chacune des instructions d'un programme.

L'unité de commande **prend en charge** une instruction se trouvant en mémoire en envoyant son adresse (flèche 7) et une commande de lecture (flèche 6) à la mémoire. Le mot d'instruction stocké dans un emplacement mémoire est alors transféré dans l'unité de commande (flèche 11). Ce mot, qui se présente sous une certaine forme binaire, est décodé par les circuits logiques de l'unité de commande qui sait maintenant de quelle instruction il s'agit. Munie de cette information, l'unité de commande génère les signaux nécessaires à l'**exécution** de l'instruction.

Unité centrale de traitement À la figure 12-2, on remarque que l'association de l'UAL et de l'unité de commande a reçu le nom de *unité centrale de traitement (UCT)*. On divise ainsi l'intérieur d'un ordinateur afin de distinguer le « cerveau » de la machine des éléments périphériques. Nous utiliserons souvent l'expression UCT dans notre description des ordinateurs, parce que cette unité, dans les micro-ordinateurs, est généralement construite dans une puce LSI, que l'on appelle le **microprocesseur.**

QUESTIONS DE RÉVISION

1. Nommez les cinq constituants de base d'un ordinateur et décrivez le rôle de chacun.
2. Que signifie UCT?
3. Que veut-on dire quand on parle de l'interfaçage avec un ordinateur.

12-5 ÉLÉMENTS DE BASE D'UN MICRO-ORDINATEUR

Il est fondamental de faire la distinction entre un micro-ordinateur et un microprocesseur. Un micro-ordinateur est constitué de plusieurs éléments dont le plus important est le microprocesseur. Ce dernier est généralement une puce intégrée renfermant tous les circuits de l'unité de commande et de l'unité arithmétique et logique — soit l'UCT. Il est donc habituel de désigner le microprocesseur comme l'unité MP, puisqu'il s'agit bien de l'unité centrale de traitement (UCT) du micro-ordinateur. C'est d'ailleurs ce qu'illustre la figure 12-3, où l'on peut voir les différents éléments de base d'un micro-ordinateur.

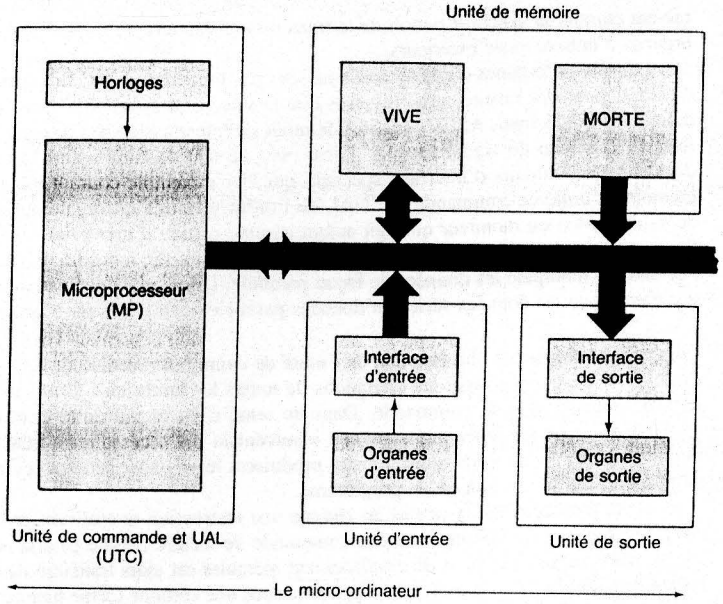


FIGURE 12-3 Éléments de base d'un micro-ordinateur.

L'unité de mémoire comprend aussi bien de la mémoire VIVE que de la mémoire MORTE. La section de la mémoire VIVE contient une ou plusieurs puces LSI montées de manière à réaliser la capacité de mémoire prévue. Cette section reçoit le programme et les données, ces dernières étant modifiées au fur et à mesure que se déroule le traitement. Cette section sert également à la mémorisation des valeurs intermédiaires et des valeurs finales des calculs réalisés pendant l'exécution du programme.

La section de la mémoire MORTE comprend une ou plusieurs puces LSI qui contiennent les instructions et les données inaltérables, par exemple les instructions du programme qui explore sans arrêt le clavier ou les données d'une table ASCII nécessaires pour sortir les informations sur un écran ou sur une imprimante.

Les sections d'entrée et de sortie renferment des circuits d'interface indispensables pour la bonne communication des organes E/S avec le reste de l'ordinateur. Dans certains cas, ces interfaces sont des puces LSI fabriquées par le constructeur du MP afin de relier ce dernier aux périphériques d'entrée/sortie. Dans d'autres cas, les circuits d'interface sont simplement des tampons.

Le Microprocesseur (MP) Le MP est le coeur de l'ordinateur. C'est lui qui a la charge des fonctions suivantes:

1. fournit les signaux de synchronisation et de commande à tous les éléments du micro-ordinateur;
2. prend en charge les instructions et les données dans la mémoire;
3. transfère les données entre la mémoire et les dispositifs E/S, et vice versa;
4. décode les instructions;
5. effectue les opérations arithmétiques et logiques commandées par les instructions;

6. réagit aux signaux de commande produits par les entrées/sorties comme le signal RESET et les INTERRUPTIONS.

Le MP renferme tous les circuits logiques nécessaires à la réalisation des fonctions précédentes, qui ne sont généralement pas accessibles de l'extérieur. Malgré cela, il est possible de contrôler d'une certaine façon ce qui se produit à l'intérieur du MP en choisissant le *programme d'instructions* que l'on place dans la mémoire et que l'on demande à l'ordinateur d'exécuter. C'est là tout le secret de la versatilité et de la souplesse d'un ordinateur — quand on veut lui demander de faire autre chose, on modifie simplement le programme placé dans la mémoire VIVE (logiciel) ou dans la mémoire MORTE (microprogramme), plutôt que de refaire le câblage interne (matériel).

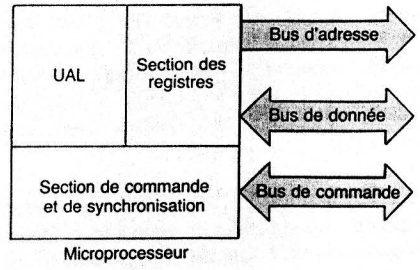
La logique interne du MP est d'une très grande complexité; pour essayer de la comprendre, on peut cependant schématiser et dire qu'elle comprend trois sections: la section *de commande et de synchronisation*, la section *des registres* et la section de l'UAL (voyez la figure 12-4). Bien qu'il y ait une interaction entre ces sections, chacune se charge de fonctions particulières.

La principale fonction de la section de commande et de synchronisation est de prendre en charge (récupérer) et de décoder (interpréter) les codes d'instruction dans la mémoire contenant le programme; à partir de l'instruction récupérée, le MP génère les signaux de commande nécessaires à ses sections pour réaliser l'exécution de l'instruction. C'est dans cette section que sont produits les signaux de commande et de synchronisation (par ex. R/W, horloge) dont ont besoin les mémoires VIVES, MORTES et les périphériques E/S.

La section des registres contient les différents registres (dans le MP) qui ont tous un rôle précis. Le plus important des registres est le **pointeur d'instructions (PI)**, qui a pour fonction de savoir quelles sont les adresses des codes d'instruction qui doivent être pris en charge dans la mémoire. Nous utiliserons le PI plus loin lorsque nous décrirons l'exécution d'un programme. Les autres registres du MP servent à des tâches du genre: conservation des codes d'instruction pendant leur décodage, conservation des données dont a besoin l'UAL, stockage des adresses des données récupérées en mémoire, et mémorisation et comptage de toutes sortes.

L'UAL, pour sa part, a pour tâche d'effectuer sur les données les opérations arithmétiques et logiques. Ces opérations sont toujours essentiellement des additions, des soustractions, des multiplications logiques (ET), des additions logiques (OU), des mises en OU-EX, des décalages, des incrémentations et des décrémentations. Dans certains MP plus sophistiqués, l'UAL renferme des circuits qui réalisent des multiplications et des divisions. Pendant une opération menée par le MP, les opérations élémentaires que l'UAL doit effectuer sont sous la régie de la section de la commande et de la synchronisation, section qui, évidemment, est tenue de respecter les ordres du code d'instruction pris en charge dans la mémoire.

FIGURE 12-4 Principales sections fonctionnelles d'une puce microprocesseur.



QUESTIONS DE RÉVISION

1. Dites la différence entre un microprocesseur et un micro-ordinateur.
2. Nommez les éléments de base d'un micro-ordinateur.
3. Nommez les principales sections d'un MP.

12-6 MOTS MACHINES

La plus petite unité d'information dans un ordinateur est le bit. Un bit isolé ne nous apprend cependant pas grand chose. C'est pour cela que l'on a choisi comme unité d'information dans un ordinateur un groupe de bits appelé un **mot**. Le nombre de bits qui constitue un mot est appelé la **longueur du mot** de l'ordinateur. La longueur d'un mot est le paramètre le plus souvent utilisé pour décrire un ordinateur. C'est ainsi que l'on désigne les ordinateurs d'après la longueur du mot qu'il traite: des ordinateurs 8 bits, des ordinateurs 16 bits... Par exemple, un ordinateur 16 bits est une machine dont les instructions des programmes et les données occupent 16 bits en mémoire et dont l'UCT traite des groupements de 16 bits. La longueur du mot sert aussi à indiquer la largeur du bus de donnée sur lequel transitent, dans les deux sens, les données entre la mémoire et l'UCT et entre l'UCT et les périphériques d'entrée/sortie.

Les plus gros ordinateurs (les grands ordinateurs et les superordinateurs) possèdent des longueurs de mot comprises entre 32 et 64 bits. La longueur du mot dans le mini-ordinateur s'échelonne entre 8 et 32 bits. Par contre, la longueur du mot dans les micro-ordinateurs se situe entre 4 et 32 bits. Les micro-ordinateurs les plus récents (par ex. PS/2 d'IBM et le Macintosh d'APPLE) utilisent des mots dont la longueur est entre 16 et 32 bits. En général, plus la longueur du mot dans un ordinateur est grande, plus cet ordinateur est performant, puisque les instructions qu'il traite et les données qu'il manipule contiennent plus d'informations. Une longueur plus grande signifie aussi un bus de donnée comprenant plus de conducteurs et donc plus d'interconnexions entre l'UCT, la mémoire et les organes d'entrée/sortie.

Nous savons déjà qu'un groupe de 8 bits s'appelle un octet. Étant donné que les micro-ordinateurs à 8 bits sont bien connus et que les éléments du code ASCII ont un octet de longueur, l'octet continue d'être cité pour décrire la longueur d'un mot et la capacité de mémoire des ordinateurs, même les plus grands. On dira d'un ordinateur 8 bits que ses mots ont une longueur d'un octet, de celui de 16 bits que ses mots ont une longueur de 2 octets... Une mémoire qui peut stocker 126 kilomots de 16 bits est décrite comme ayant une capacité de 256 kilo-octets.

Types de mots machines Un mot conservé dans une mémoire d'ordinateur peut correspondre à deux types d'information — à une *instruction* ou à une *donnée*. Les données sont des nombres ou des caractères que le programme de l'UCT soumet à diverses opérations commandées par le programme. Les données se présentent sous diverses formes: binaires signées, non signées, DCB, à virgule flottante (comme dans la notation technique), codes ASCII (dans le cas des caractères). Voici l'exemple du nombre $+86_{10}$ tel qu'il apparaît quand il est mémorisé dans un mot de 8 bits:

01010110

Autre exemple, voici à quoi ressemble dans un mot de 8 bits, le code ASCII de la lettre « V ».

01010110

Remarquez que ces deux codes sont identiques. L'ordinateur ne sait pas faire la différence entre les deux. C'est la tâche du programmeur de savoir quel type de donnée est mémorisé et de s'assurer que son programme fait la bonne traduction et traite les données correctement.

Dernier exemple, voici un mot de 16 bits qui conserve deux caractères ASCII:

$$\underbrace{0101011001010111}_{\text{V}\qquad\text{W}}$$

Le même mot de 16 bits pourrait servir à représenter le nombre $+22103_{10}$. Évidemment, les mots plus longs sont en mesure de mémoriser plus de caractères et des nombres plus grands.

Les mots d'instruction sont plus complexes que les mots de donnée. La section suivante traite des premiers.

QUESTIONS DE RÉVISION

1. Quels sont les deux types d'information qui se retrouvent dans les mots machines?
2. Quel est l'avantage de disposer d'un ordinateur ayant un mot plus long.?

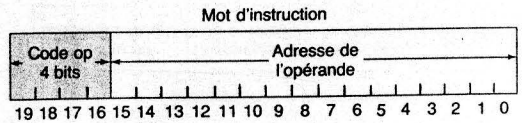
12-7 MOTS D'INSTRUCTION

Le format retenu pour les *données* varie peu entre ordinateurs différents, surtout s'ils utilisent la même longueur de mot. C'est tout le contraire pour les mots d'instruction. Ces mots contiennent tout ce qui est nécessaire à un ordinateur pour réaliser ses diverses opérations, et le format et les codes employés varient grandement d'un ordinateur à l'autre. Suivant l'ordinateur, les informations consignées dans le mot d'instruction pourront différer. Malgré tout et pour la plupart des machines, les mots d'instruction précisent deux éléments d'information de base: l'*opération* à effectuer et l'*adresse* de l'*opérande* (donnée) qui est l'objet de l'opération.

La figure 12-5 illustre l'exemple d'un *mot d'instruction à une adresse* se trouvant dans un ordinateur à 20 bits hypothétique. Les 20 bits du mot d'instruction sont divisés en deux parties. La première partie du mot (bits 16 à 19) contient le **code d'opération** (ou **code op** en abrégé). Ce code de 4 bits représente l'opération demandée à l'ordinateur, par exemple une addition ou une soustraction. La seconde partie (les bits 0 à 15) représente l'**adresse de l'opérande**, soit l'endroit dans la mémoire où est conservé l'opérande.

En utilisant 4 bits pour le code d'opération, on conclue qu'il y a $2^4 = 16$ codes op possibles, chacun correspondant à une instruction différente. Cela signifie qu'un ordinateur qui fonctionne avec un mot d'instruction de cette nature ne peut supporter que 16 instructions différentes. Un ordinateur plus versatile doit avoir un plus grand nombre d'instructions et doit, de ce fait, réserver plus de bits pour son code d'opération. Quoiqu'il en soit, chaque instruction a un code d'opération que l'ordinateur (unité de commande) doit interpréter (décoder).

FIGURE 12-5 Mot d'instruction à une adresse.



Le mot d'instruction de la figure 12-5 a réservé 16 bits pour le code de l'adresse de l'opérande. Avec 16 bits, on peut constituer $2^{16} = 65\,536$ adresses possibles. En conclusion, ce mot d'instruction précis peut décrire 16 opérations différentes et retrouver 65 536 adresses d'opérande. À titre d'exemple, un mot de 20 bits pourra contenir

0 1 0 0	0 1 1 1 1 0 1 0 0 1 1 1 0 0 1 0
Code op	Code d'adresse

Le code op 0100 représente l'une des 16 opérations possibles implantées dans la machine; supposons qu'il s'agit de l'opération d'*addition* (ADD). Le code d'adresse est 0101101001110010, ou 5A72 en hexadécimal (voyez comment cette dernière notation simplifie les choses). En fait, l'instruction au complet se lit comme ceci en hexadécimal:

4	5 A 7 2
Code op	Adresse

Cette instruction au complet signale à l'ordinateur de faire ceci:

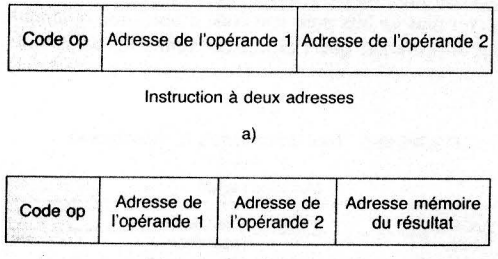
Prends en charge le mot stocké dans l'emplacement mémoire 5A72, envoie-le dans l'UAL et additionne-le au nombre se trouvant dans le registre accumulateur. La somme est ensuite conservée dans l'accumulateur. (Le contenu précédent de l'accumulateur est perdu.)

Nous examinerons cette instruction, ainsi que d'autres, dans une section ultérieure.

Instructions à adresses multiples L'instruction à une adresse dont nous venons de parler a déjà été le principal type d'instruction que l'on retrouvait dans les ordinateurs. Les grands ordinateurs, les premiers, ont par la suite commencé à utiliser des formats de mots d'instruction contenant plus d'informations par mot.

La figure 12-6 reproduit deux formats de mot d'instruction qui comprennent plus d'une adresse. L'instruction à deux adresses possède un code op et les adresses des *deux* opérandes traités par l'opération indiquée. Par ailleurs, le mot d'instruction à trois adresses possède, en plus du code op, les deux adresses des opérandes et l'adresse en mémoire où placer le résultat.

FIGURE 12-6 Formats d'instructions à adresses multiples.



Ces mots d'instruction à adresses multiples possèdent l'avantage manifeste de contenir plus de renseignements que celle à une seule adresse. Il faut en conclure qu'un ordinateur qui travaille avec des mots à plusieurs adresses requière moins d'instructions pour accomplir une tâche précise demandée par le programme. Évidemment, des mots d'instruction plus longs signifient une mémoire de plus grande capacité.

Instructions multioctets Nous venons de voir des formats de mot d'instruction qui contenaient le code op et l'adresse de l'opérande dans un *seul* mot. Autrement dit, une instruction complète, comme celles illustrées dans les figures 12-5 ou 12-6, qui tient dans un *seul* emplacement de la mémoire. Il s'agit là du cas typique des grands ordinateurs qui travaillent avec de longs mots. En ce qui concerne les micro-ordinateurs et les mini-ordinateurs, le fait d'utiliser des mots plus courts rend impossible d'avoir le code op et l'adresse de l'opérande dans un seul mot à la fois.

Comme un assez bon nombre de micro-ordinateurs travaillent avec des mots de 8 bits (un octet) de longueur, nous allons décrire les formats des instructions pour les ordinateurs 8 bits. Au moyen d'instructions ayant un seul octet de longueur, on constitue *trois* formats différents: les instructions à un octet, celles à deux octets et celles à trois octets. Voyez à la figure 12-7 la structure de ces trois types.

Il y a d'abord l'instruction à un octet qui contient un code op de 8 bits et aucun bit pour l'adresse. On s'aperçoit bien qu'une instruction de ce genre ne stipule aucune donnée en mémoire que traite cette instruction. Pour cette raison, ces instructions sont réservées aux instructions qui n'ont pas besoin de récupérer des données en mémoire. Un exemple

FIGURE 12-7 Formats des instructions dans les micro-ordinateurs 8 bits.

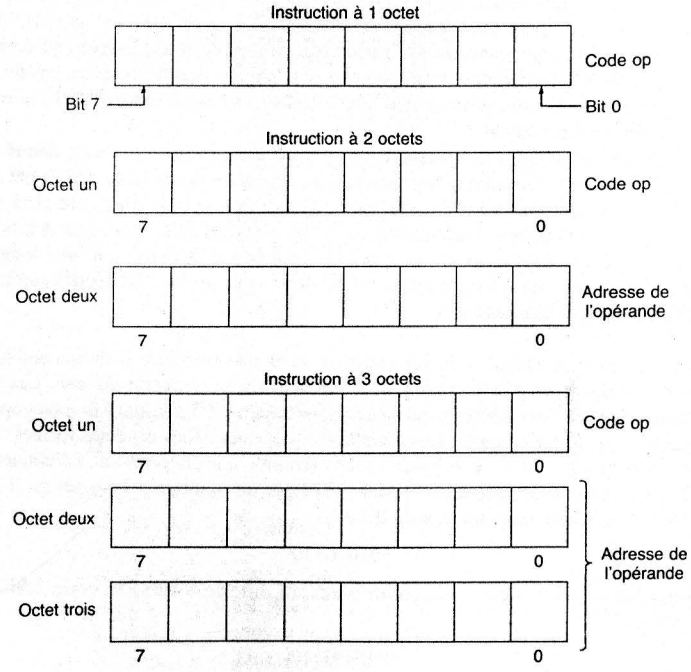


TABLEAU 12-1

Adresse mémoire (Hex)	Mot mémoire Binaire	Hex	Description
0020	01001010	4A	Code op de ADD
0021	11110110	F6	Bits d'adresse de rangs inférieurs (LO)
0022	00110101	35	Bits d'adresse de rangs supérieurs (HI)
.	.		
.	.		
.	.		
.	.		
.	.		
35F6	01111100	7C	Opérande

d'une telle instruction est l'opération *remettre l'accumulateur à zéro* (CLA), qui commande à l'ordinateur de mettre à zéro toutes les sorties des bascules de l'accumulateur de l'UAL.

Le premier octet de l'instruction à deux octets est un code op et le second, un code d'adresse de 8 bits qui indique l'emplacement en mémoire où se trouve l'opérande. Dans une instruction à trois octets, les deuxième et troisième octets se combinent pour former une adresse d'opérande de 16 bits. Pour ces instructions multioctets, les deux ou trois octets qui constituent l'instruction complète sont mémorisés dans des emplacements en mémoire successifs. C'est d'ailleurs ce qu'illustre le tableau 12-1 pour le cas d'une instruction à trois octets. La colonne de gauche donne la liste des emplacements en mémoire où se trouve chaque octet du mot. Ces adresses sont exprimées en hexadécimal. La deuxième colonne reproduit le mot binaire tel qu'il apparaît dans la case mémoire; la troisième colonne est l'équivalent en hexadécimal du mot. Examinez bien cet exemple dans tous ses détails avant de poursuivre et d'essayer de trouver la signification de ces chiffres.

Les trois octets mémorisés dans les emplacements 0020, 0021, 0022 forment tous ensemble l'instruction complète, soit l'addition de la donnée dans l'adresse 35F6 au nombre dans l'accumulateur. Les deuxième et troisième octets renferment les 8 bits de rangs inférieurs (LO) et les 8 bits de rangs supérieurs, respectivement, de l'adresse de l'opérande. Certains micro-ordinateurs fonctionnent selon l'ordre inverse, c'est-à-dire les bits supérieurs sont dans le deuxième octet et les bits inférieurs dans le troisième octet. L'emplacement mémoire 35F6 est également indiqué; cette case mémoire contient la donnée 7C, soit la donnée prise en charge par l'unité de commande et envoyée à l'UAL en vue de son addition au contenu de l'accumulateur.

QUESTIONS DE RÉVISION

1. Qu'est-ce-que le code op?
2. Définissez adresse de l'opérande.
3. Quelle information est contenue dans le premier octet d'une instruction multioctets?

12-8 EXÉCUTION D'UN PROGRAMME EN LANGAGE MACHINE

Les mots d'instruction que nous venons de décrire ont reçu le nom d'instructions du **langage machine**, parce qu'on n'y trouve que des 0 et des 1, les deux seuls états que l'ordinateur peut comprendre. De nombreux autres langages peuvent servir pour la programmation d'un ordinateur, et vous en connaissez sûrement quelques-uns, comme le BASIC ou le PASCAL.

TABLEAU 12-2

Mnémonique	Code op Binaire/Hex	Description du fonctionnement
LDA	01011011/5B	*Charge l'accumulateur:* La donnée mémorisée dans l'adresse de l'opérande est chargée dans l'accumulateur.
ADD	01101101/6D	*Additionne:* L'opérande est additionné au nombre contenu dans l'accumulateur et la somme devient le nouveau nombre dans l'accumulateur.
SUB	10100010/A2	*Soustrait:* L'opérande est retranché du contenu de l'accumulateur et la différence devient le nouveau nombre dans l'accumulateur.
STA	10001010/8A	*Charge l'accumulateur:* Le contenu de l'accumulateur est chargé en mémoire à l'adresse spécifiée pour l'opérande.
JMP	0100100/4C	*Branchement (non conditionnel):* L'instruction suivante est trouvée à l'endroit indiqué par l'adresse de l'opérande, et non dans une case mémoire qui suit.
JPZ	0110110/76	*Branchement si zéro:* L'instruction suivante est trouvée dans l'adresse de l'opérande si le contenu de l'accumulateur est *zéro*. Autrement, elle est dans la case mémoire qui suit.
HLT	0011111/3F	*Arrêt:* L'ordinateur cesse son travail. Aucune autre instruction n'est plus traitée.

Ces **langages évolués** ont été conçus pour faciliter le processus de rédaction d'un programme. Il est essentiel que vous compreniez que l'emploi d'un langage évolué nécessite quand même la conversion du programme en langage machine et la mémorisation de la version machine ainsi obtenue dans la mémoire de l'ordinateur avant le début de l'exécution du programme. Dans un micro-ordinateur courant, la conversion du BASIC au langage machine est l'oeuvre d'un programme dédié conservé dans de la mémoire MORTE.

Pour illustrer le déroulement d'un programme machine dans un ordinateur, nous utiliserons les instructions répertoriées dans le tableau 12-2. Ces instructions n'appartiennent à aucun ordinateur en particulier, mais sont représentatives des instructions qu'on peut retrouver dans la plupart des micro-ordinateurs. Chaque instruction est repérée par un **mnémonique**, c'est-à-dire une abréviation qui nous rappelle le code d'opération. Lisez dans la colonne de droite la description de chacune de ces instructions et n'oubliez jamais que l'opérande dont il est question est la donnée se trouvant à l'adresse indiquée dans l'instruction.

Nous allons nous servir de quelques-unes des instructions du tableau 12-2 pour rédiger un programme en langage machine dont l'adresse de début est 0000 et qui accomplit ce qui suit:

1. soustraire un mot de 8 bits (X) d'un autre mot de 8 bits (Y). Les valeurs X et Y se trouvant mémorisées dans les emplacements dont les adresses en hexadécimal sont 0300 et 0301, respectivement;
2. mémorise la différence (DIF) dans l'emplacement mémoire dont l'adresse est 0302;
3. si la différence est zéro, le programme est dérouté vers l'adresse 0400 où se trouve sa prochaine instruction. Autrement le programme s'arrête.

TABLEAU 12-3

Adresse mémoire (Hex)	Mot en mémoire (Hex)	Mnémonique	Description
0000 0011 0002	5B 00 03	LDA$0300	Charge la donnée (Y) de l'adresse 0300 dans l'accumulateur.
0003 0004 0005	A2 01 03	SUB$0301	Soustrait la donnée (X) se trouvant dans l'adresse de l'accumulateur.
0006 0007 0008	8A 02 03	STA$0302	Mémorise la différence dans la case mémoire ayant l'adresse 0302.
0009 000A 000B	76 00 04	JPZ$0400	Si DIF est zéro, se branche sur l'adresse 0400 où se trouve l'instruction suivante.
000C	3F	HLT	Si DIF est *non* nul, le programme s'arrête.
.	.	.	.
.	.	.	.
.	.	.	.
0300	??	Y	Adresse de la donnée Y.
0301	??	X	Adresse de la donnée X.
0302	??	DIF	Adresse de la case où est mémorisée la différence.

Le tableau 12-3 reproduit la totalité de ce programme tel qu'il se présente dans la mémoire de l'ordinateur. En réalité, le programme machine proprement dit correspond aux deux premières colonnes. La première colonne liste les adresses en hexadécimal de tous les emplacements mémoires rejoint par ce programme. La deuxième colonne donne le code hexadécimal du mot stocké à l'adresse correspondante. Rappelez-vous que ces nombres hexadécimaux remplacent simplement les adresses et codes d'instruction en binaire qui figurent réellement dans l'ordinateur (il ne peut décoder que des valeurs binaires).

La troisième colonne contient le mnémonique ainsi que l'adresse (s'il y a lieu) de l'instruction. La dernière colonne décrit l'opération effectuée par l'instruction. Par exemple, la première instruction du programme en est une de trois octets. Le premier octet contient l'adresse mémoire où se trouve le code op 5B. Les deuxième et troisième octets gardent les adresses 0001 et 0002 dont la combinaison des contenus permet d'obtenir l'adresse de l'opérande (0300). Le mnémonique de cette instruction est LDA $0300. L'expression LDA signifie « charge l'accumulateur », et $0300 est l'adresse de l'opérande. Le signe $ sert souvent à indiquer que l'adresse est exprimée en hexadécimal. Quand l'ordinateur exécute cette instruction, il lit la donnée Y mémorisée à l'adresse 0300 et la charge dans l'accumulateur.

Le programme commence à l'adresse 0000 et prend fin à l'adresse 000C. Les emplacements référencés par les adresses 0300 et 0302 sont utilisés pour conserver des données. Le contenu de ces emplacements est inconnu, car on y met habituellement des variables. Les emplacements dont les adresses vont de 000C à 0300 ne servent pas.

Exécution du programme Nous allons à présent parcourir pas à pas chaque étape de l'exécution du programme par l'ordinateur. Notre intention ici est de donner un aperçu des

fonctions principales sans se soucier des détails de tout ce qui se déroule à l'intérieur de la machine. Tout particulièrement, nous allons constater que le travail de l'ordinateur implique toujours deux genres de cycle de fonctionnement: 1) un cycle de **prise en charge** au cours duquel l'unité de commande récupère les codes d'instruction (le code op et l'adresse de l'opérande) dans la mémoire; 2) un **cycle d'exécution** au cours duquel l'unité de commande réalise l'opération définie par le code op.

Le fonctionnement débute au moment où l'opérateur appuie sur l'interrupteur DÉBUT ou MARCHE. Cela a pour effet d'initialiser le pointeur d'instructions avec la valeur 0000. Le pointeur d'instructions (PI), registre de l'unité de commande, fait le suivi des adresses du programme à mesure que l'ordinateur exécute les instructions.

1. L'unité de commande prend en charge le premier octet de l'adresse 0000, selon l'indication du PI. Cet octet vaut 5B, soit le code op de la première instruction. Les circuits de l'unité de commande évaluent que ce code op particulier signifie LDA (charge l'accumulateur) et que l'adresse de l'opérande doit suivre ce code.
2. L'unité de commande incrémente le PI à 0001 et prend en charge l'octet conservé dans la case de cette nouvelle adresse. L'octet récupéré est 00, soit l'octet de rang inférieur de l'adresse de l'opérande. L'unité de commande incrémente le PI à 0002 et prend en charge l'octet dans la case mémoire référencée par cette nouvelle adresse. Cet octet vaut 03 et c'est l'octet de rang supérieur de l'adresse de l'opérande.
3. Cela met un terme au cycle de prise en charge; l'unité de commande dispose maintenant du code op et de l'adresse de l'opérande (0300) nécessaire pour exécuter l'instruction LDA.
4. L'unité de commande mène à bien l'ordre indiqué par cette instruction en lisant le mot dans l'adresse 0300 et en plaçant ce dernier dans l'accumulateur. Ceci termine le cycle d'exécution.
5. Le PI est incrémenté jusqu'à 0003 et l'unité de commande amorce un nouveau cycle de prise en charge en récupérant le code op qui se trouve à cette adresse. L'unité de commande s'aperçoit qu'il s'agit de l'instruction A2, soit une opération de soustraction; il sait qu'avec cette instruction vient l'adresse d'un opérande, adresse stockée dans les deux octets suivants du programme.
6. L'unité découvre l'adresse de l'opérande 0301 de la même façon qu'à l'étape 2. ci-dessus. La valeur dans le PI est maintenant 0005.
7. Cela termine le cycle de prise en charge et l'unité de commande contient à présent le code op et l'adresse de l'opérande qui lui permettent de mener à bien l'exécution de l'instruction SUB. L'unité de commande réalise cette instruction en lisant la donnée contenue dans la case dont l'adresse est 0301 et en l'envoyant dans l'UAL qui soustraira cette donnée du contenu de l'accumulateur.
8. Le PI est incrémenté jusqu'à 0006 afin de lancer un nouveau cycle de prise en charge. L'unité de commande récupère dans la mémoire le code 8A se trouvant dans l'adresse du PI; elle reconnaît qu'il s'agit de l'opération STA et elle s'attend à trouver l'adresse de l'opérande dans les deux octets suivants du programme.
9. L'unité de commande prend en charge l'adresse de l'opérande 0302. Le PI contient à présent 0008. Le contenu de l'accumulateur est transféré dans la case mémoire ayant l'adresse 0302.
10. Le PI est incrémenté jusqu'à 0009 en vue de lancer le prochain cycle de prise en charge, l'unité de commande retrouve le code op 76 se trouvant dans cette adresse. Ce code est identifié comme étant une opération JPZ; l'unité sait que

pour cette opération, une décision de branchement est prise d'après la valeur se trouvant dans l'accumulateur.

11. Si la donnée dans l'accumulateur est exactement zéro, l'unité de commande calcule l'adresse de l'opérande à partir des valeurs dans les deux adresses suivantes (000A et 000B). Cette adresse d'opérande est chargée dans le PI, ce qui signale à l'unité de commande que la prochaine instruction du programme se trouve dans l'adresse 10400 et que le programme poursuit son déroulement à partir de cette adresse.
12. Par contre, si la donnée dans l'accumulateur n'est pas zéro, l'unité de commande incrémente trois fois le PI qui passe à 000C. L'instruction suivante qui est prise en charge se trouve dans cette adresse; on reconnaît qu'il s'agit de l'instruction HLT, dont la fonction est de faire cesser les instructions de prise en charge et d'exécution.

Ce petit programme simple met en évidence bien des principes que l'on retrouve dans les programmes en langage machine exécutés par un ordinateur; malgré cela, rien dans ce programme ne dévoile le moindrement les énormes capacités et possibilités d'un ordinateur. Il est important que le lecteur comprenne bien qu'un programme est exécuté séquentiellement à partir d'une adresse de départ (0000 par exemple). La séquence des adresses peut être modifiée par des instructions de branchement comme JPZ ou JMP (étapes 10 et 11 de ci-dessus). Même si un programme est exécuté pas à pas, l'ordinateur fonctionne tellement rapidement que le programme sort ses résultats en très peu de temps. Ainsi, notre petit programme serait effectué en moins de 20 μs même sur le micro-ordinateur le moins rapide.

QUESTIONS DE RÉVISION

1. Dites quelle est la différence entre un programme en langage machine et un programme en langage évolué.
2. Définissez le mnémonique d'une instruction.
3. Quelles actions ont généralement lieu durant le cycle de prise en charge? Le cycle d'exécution?
4. Quelle est la fonction du pointeur d'instructions?

12-9 STRUCTURE TYPE D'UN MICRO-ORDINATEUR

Nous avons maintenant tout ce qu'il nous faut pour examiner en détail l'organisation d'un micro-ordinateur. Les nombreuses architectures possibles pour un micro-ordinateur sont essentiellement les mêmes, les distinctions portant sur des éléments comme la longueur des données et la largeur des bus d'adresse, les types de signaux de commande utilisés. Afin de procéder de la façon la plus claire possible, il convient de s'arrêter sur une seule architecture et de l'étudier en détail. Une fois que nous connaîtrons bien ce micro-ordinateur type, il deviendra relativement aisé d'étendre nos connaissances à un autre type. La structure de micro-ordinateur que nous avons choisie en est une très courante de nos jours; voyez-en l'illustration à la figure 12-8.

Ce schéma nous fait voir les constituants de base d'un micro-ordinateur de 8 bits, ainsi que les divers bus qui connectent tous ces constituants ensemble. Malgré sa complexité, ce diagramme n'expose pas tous les détails d'un micro-ordinateur. Pour nos besoins, cela suffit amplement.

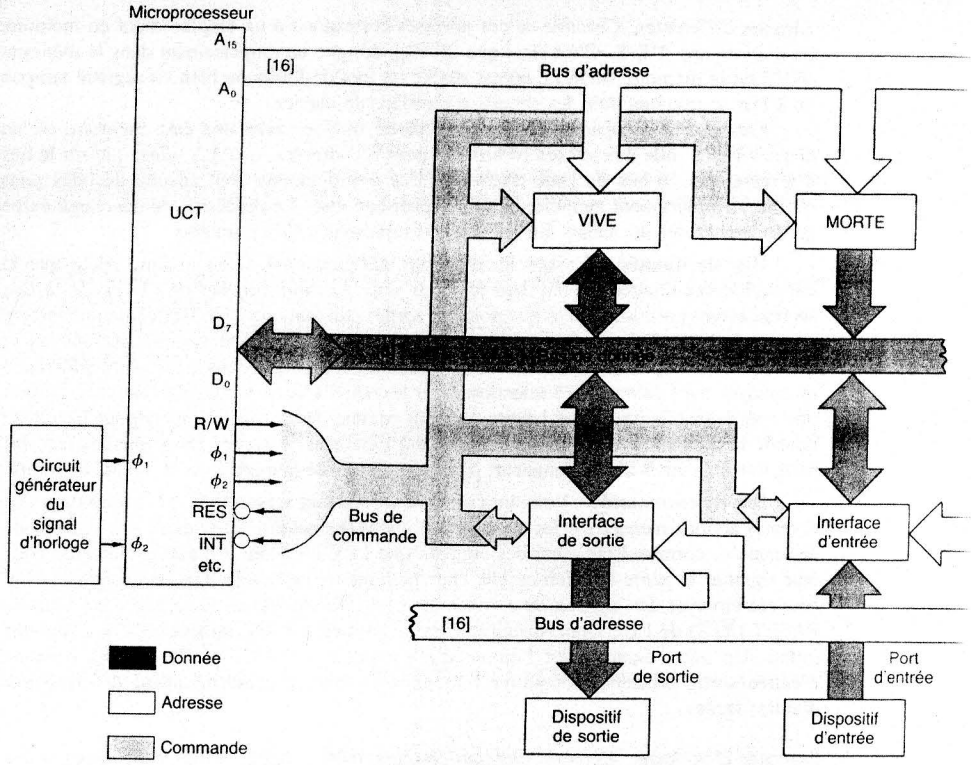


FIGURE 12-8 Structure type d'un micro-ordinateur de 8 bits.

Le système des bus Dans un micro-ordinateur, il y a trois bus qui véhiculent l'information et les signaux nécessaires à la bonne marche du système. Ces bus raccordent le microprocesseur (UCT) à chaque type de mémoire, aux organes E/S; c'est grâce à eux que les données et l'information parviennent à circuler entre l'UCT et les autres éléments. Sommairement, on peut dire que l'UCT passe son temps à recevoir des données des divers éléments comme la mémoire, les entrées et les sorties et à leur en envoyer.

Dans un micro-ordinateur, l'UCT est tenue au courant de tous les transferts d'information. Quand l'UCT envoie des données à un autre dispositif informatique, elle réalise une opération d'écriture qui, dans les faits, consiste à placer une donnée dans l'élément sélectionné. Quand l'UCT reçoit une donnée d'un autre élément, ce qu'on appelle une opération de lecture, elle récupère cette donnée de l'élément sélectionné. Il est important de bien réaliser que derrière les expressions « lecture » et « écriture », se dissimulent de véritables transferts de données avec l'UCT.

Voici le rôle des différents bus du micro-ordinateur durant les transferts de données:

Bus d'adresse Il s'agit d'un bus *unidirectionnel*, puisque les informations qui y circulent vont toujours dans un même sens, soit de l'UCT vers la mémoires ou les dispositifs E/S. Seule l'UCT peut placer des niveaux logiques sur les conducteurs du bus d'adresse; comme il y a 16 conducteurs, dans ce cas-ci, l'UCT peut atteindre jusqu'à $2^{16} = 65\,536$

adresses différentes. Chacune de ces adresses correspond à un emplacement en mémoire ou à un organe d'E/S. C'est ainsi que $20A0_{16}$ désigne un emplacement dans la mémoire VIVE ou la mémoire MORTE où est stocké un mot de 8 bits ou bien un registre tampon de 8 bits se trouvant dans les circuits d'interface du clavier.

Quand l'UCT désire communiquer (écrire ou lire) avec une case mémoire ou un dispositif E/S, elle met sur ses 16 broches pour les adresses A_0 à A_{15}, ainsi que sur le bus d'adresse, les 16 bits du code d'adresse. Ces bits d'adresse sont ensuite décodés pour obtenir l'emplacement mémoire ou le périphérique visé. Le processus de décodage exige généralement des décodeurs qui ne sont pas représentés sur ce schéma.

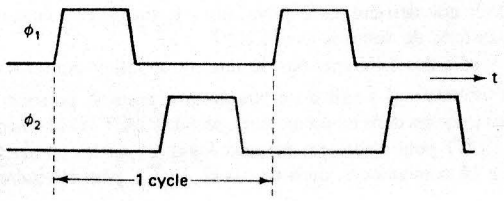
Bus de donnée Il s'agit ici d'un bus *bidirectionnel*, ainsi nommé parce que la circulation des données se fait dans les deux sens. Les huit broches de l'UCT, D_0 à D_7, servent autant pour les entrées que pour les sorties, suivant que l'UCT effectue une lecture ou une écriture. Durant une opération de lecture, elles agissent comme des entrées et reçoivent la donnée se trouvant sur le bus de donnée et prélevée dans la mémoire référencée ou envoyée par l'élément E/S sélectionné par le code d'adresse sur le bus d'adresse. Durant une opération d'écriture, ces bornes agissent comme des sorties et mettent sur le bus de donnée la donnée envoyée à la mémoire ou à l'élément de sortie. Dans tous les cas, les mots transmis ont 8 bits de longueur, parce que l'UCT ne manipule que des mots de 8 bits.

Bus de commande Voici un ensemble de signaux servant à la synchronisation des diverses actions menées par les éléments du micro-ordinateur. Certains de ces signaux de commande, comme $R/\overline{W}$, sont des signaux que l'UCT envoie à d'autres éléments pour leur signifier le genre d'action qu'elle cherche à mettre en branle. Les périphériques E/S peuvent envoyer des signaux de commande à l'UCT. Un signal de ce genre est l'entrée RESET ($\overline{RES}$) de l'UCT qui, quand elle passe au niveau BAS, ramène l'UCT à son état initial. Un autre exemple est l'entrée d'interruption de l'UCT ($\overline{INT}$) que les organes d'entrée/sortie utilisent pour attirer l'attention du microprocesseur quand il travaille à d'autres tâches.

Signaux d'horloge Certains microprocesseurs (8085, 6802, 6502) font appel à un système de cadencement à deux phases sans chevauchement, comme ce qu'illustre la figure 12-9. Par contre, d'autres microprocesseurs comme le Z-80 et le 8086 ont un fonctionnement qui repose sur un seul signal d'horloge. Dans nos discussions à venir, nous allons examiner un système d'horloge biphasé. Les deux phases de l'horloge, ϕ_1 et ϕ_2, sont toujours présentes dans le bus de commande. D'autres signaux de synchronisation, dérivés de ces deux derniers, sont parfois générés par l'UCT et sont aussi présents dans le bus de commande.

Ports E/S Durant l'exécution d'un programme, l'UCT effectue sans arrêt des lectures et des écritures en mémoire. En plus, le programme pourra demander à cette unité de lire un

FIGURE 12-9 Système d'horloge biphasé.



dispositif d'entrée ou d'écrire dans un dispositif de sortie. Sur le diagramme de la figure 12-8 représentant un micro-ordinateur de 8 bits, on ne voit qu'un dispositif d'entrée et qu'un dispositif de sortie; dans la réalité, il peut y avoir plusieurs dispositifs d'entrée et autant de dispositifs de sortie. Chacun étant doté d'un circuit d'interface qui le raccorde au système de bus du micro-ordinateur. Comme on l'a appris, ces interfaces ont pour fonction d'assurer la compatibilité entre le dispositif et le micro afin que des données puissent être échangées entre eux. Chaque fois que le dispositif E/S utilise des niveaux de tension, des signaux de synchronisation ou des formats de signaux différents de ce qui a cours dans l'UCT, il est indispensable d'avoir une interface.

Par exemple, l'unité d'entrée/sortie classique, le téléscripteur, (abréviation TTY) envoie des codes ASCII à un ordinateur sur une voie *série* (1 bit à la fois sur un seul conducteur). Toutefois, le microprocesseur n'accepte que des données sous forme *parallèle* sur son bus de donnée de 8 bits. Par conséquent, il faut donc un circuit d'interface qui convertit les signaux série du TTY en un mot de données de 8 bits parallèles et un autre circuit d'interface qui transforme la sortie parallèle de l'UCT en un signal série compréhensible par le TTY.

Même si jusqu'à présent nous avons considéré les éléments E/S comme des emplacements mémoires, il en diffère par plusieurs aspects. La principale différence étant que ces éléments ont la capacité d'interrompre le travail de l'UCT. Qu'est-ce que cela signifie: et bien, le dispositif peut envoyer à l'unité de traitement central un signal d'avertissement $\overline{INT}$ qui exprime son désire de communiquer avec lui. L'unité met en suspend son exécution actuelle et entreprend de répondre à la demande du périphérique interrupteur. Les mémoires VIVES et MORTES n'ont pas la possibilité d'envoyer des signaux d'interruption.

QUESTIONS DE RÉVISION

1. Donnez une description des fonctions des trois bus que l'on retrouve dans un micro-ordinateur standard.
2. Quel bus est unidirectionnel?
3. Quel bus renferme les signaux d'horloge?

12-10 OPÉRATIONS DE LECTURE ET D'ÉCRITURE

Nous sommes maintenant prêt à examiner de façon détaillée les communications de l'UCT avec les autres éléments du microprocesseur. Rappelez-vous que l'unité centrale de traitement renferme tous les circuits en mesure d'exécuter les instructions d'un programme conservé dans la mémoire VIVE ou la mémoire MORTE. L'UCT effectue continuellement des opérations de lecture et d'écriture pendant l'exécution d'un programme. Chaque fois qu'elle prend en charge une instruction, elle réalise une opération de lecture. Après chaque décodage d'une instruction, elle doit effectuer une autre lecture de la mémoire pour récupérer l'opérande de l'instruction ou bien procéder à une écriture dans une case mémoire. Dans certains cas, l'instruction demande à l'UCT de lire une donnée fournie par un dispositif d'entrée (comme un clavier ou un CAN) ou d'écrire une donnée sur un périphérique (comme un affichage à DEL ou une unité de disque magnétique).

L'opération de lecture Voici la suite des étapes se produisant durant une opération de lecture.

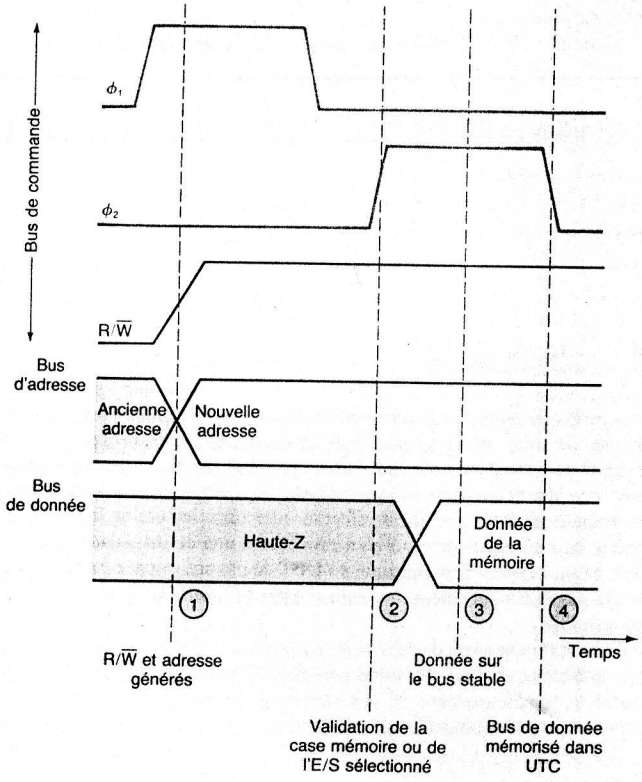
1. L'UCT génère le niveau logique approprié sur la ligne $R/\overline{W}$ qui amorce l'opération de lecture . Normalement $R/\overline{W} = 1$ pour la lecture. Le conducteur

de ce signal fait partie intégrante du bus de commande et aboutit à toutes les mémoires et à tous les périphériques E/S.

2. Simultanément, l'unité centrale place sur le bus d'adresse le code d'adresse de 16 bits qui désigne l'emplacement mémoire ou l'élément d'entrée ou de sortie qui doit envoyer une donnée à l'UCT.
3. La case mémoire ou le périphérique sélectionné place sur le bus de donnée un mot de 8 bits. Toutes les autres cases mémoires ou périphériques non sélectionnés ne sont pas du tout touchés par cette opération, car leurs sorties *trois états* restent dans l'état haute impédance (état désactivé).
4. L'UCT recueille sur ses broches D_0 à D_7 le mot de 8 bits amené par le bus de donnée. Ces broches de données agissent comme des entrées chaque fois que $R/\overline{W} = 1$. Le mot de 8 bits est alors mémorisé dans l'un des registres de l'unité centrale de traitement, par exemple l'accumulateur.

Cette séquence se comprend mieux à l'aide du chronogramme illustré à la figure 12-10 qui met en évidence les rapports entre les signaux des divers bus. Toutes les actions qui se produisent sont rapportées aux signaux d'horloge ϕ_1 et ϕ_2. L'opération de lecture intégrale a lieu pendant un cycle d'horloge. Ce qui correspond normalement à une

FIGURE 12-10 Synchronisation caractéristique d'une opération de lecture.



durée entre 100 ns et 1 μs pour les microprocesseurs MOS. Le front montant de ϕ_1 déclenche la production par l'UCT des signaux $R/\overline{W}$ et d'adresse. Après un court délai, la ligne $R/\overline{W}$ passe au niveau HAUT et le bus d'adresse contient désormais la nouvelle adresse (point 1 sur le chronogramme). Notez que la forme d'onde du bus d'adresse illustre les deux transitions possibles (BAS à HAUT et HAUT à BAS), pour la simple raison que parmi les 16 conducteurs, certains vont effectuer la transition dans un sens et d'autres dans l'autre sens.

Durant l'impulsion ϕ_2, la case mémoire ou le périphérique E/S sélectionné est validé (point 2) et se prépare à installer son mot sur le bus de donnée. Avant cette action, le bus de donnée se trouve dans l'état haute impédance, puisqu'aucun dispositif qui lui est relié n'est validé. À un certain instant de l'impulsion ϕ_2, la donnée placée sur le bus devient stable (point 3). De nouveau, on illustre sur le diagramme les deux transitions possibles. Le retard entre le début de l'impulsion ϕ_2 et le moment où la donnée se stabilise sur le bus est fonction de la vitesse de réponse de la mémoire ou des éléments E/S. Dans le cas de la mémoire, ce retard est son *temps d'accès*. À l'instant du front descendant de ϕ_2, la donnée sur le bus de donnée est mémorisée dans l'UCT (point 4). On s'aperçoit bien que la mémoire et les périphériques E/S doivent être en mesure de placer leur donnée sur le bus avant le front descendant de ϕ_2, faute de quoi le transfert à l'unité centrale de traitement n'a pas lieu. Il est donc primordial de s'assurer que la vitesse des dispositifs est compatible avec la fréquence d'horloge du micro-ordinateur.

L'opération d'écriture Les étapes suivantes interviennent au moment d'une opération d'écriture.

1. L'UCT génère le niveau logique qui permet à la ligne $R/\overline{W}$ d'amorcer une opération d'écriture. Normalement, $R/\overline{W} = 0$ pour une opération d'écriture.
2. Au même instant, l'UCT place un code d'adresse de 16 bits sur le but d'adresse.
3. L'unité de traitement place ensuite un mot de 8 bits sur le bus de donnée en utilisant ses bornes D_0 à D_7, mais cette fois-ci comme sorties . Ce mot de 8 bits provient le plus souvent d'un registre de l'unité de traitement, comme l'accumulateur. Tous les autres dispositifs raccordés au bus de donnée ont leurs sorties désactivées.
4. La case mémoire ou le périphérique E/S sélectionné recueille la donnée sur le bus de donnée. Toutes les cases et éléments E/S non sélectionnés ont leurs entrées invalidées.

Cette séquence respecte le chronogramme de la figure 12-11. De nouveau, on remarque que le front montant de ϕ_1 déclenche les signaux $R/\overline{W}$ et du bus d'adresse (point 1). Pendant l'impulsion ϕ_2, la case mémoire sélectionnée ou le périphérique E/S est invalidé (point 2) et l'UCT place sa donnée sur le bus de donnée. Les niveaux électriques du bus de donnée se stabilisent peu de temps après le début de l'impulsion ϕ_2. La donnée est donc écrite à l'endroit spécifié (en mémoire ou dans un dispositif) pendant que ϕ_2 est à 1. Si c'est un dispositif E/S qui a été choisi, la mémorisation de la donnée transportée par le bus de donnée se produit à l'instant du front descendant de cette impulsion (point 4).

Les opérations de lecture et d'écriture résument presque la totalité des activités qui se produisent à l'extérieur de l'unité centrale de traitement. Voici un exemple qui démontre bien ceci:

EXEMPLE 12-1

Ci-dessous, nous reproduisons un petit programme stocké dans les cases mémoires 0020_{16} à 0029_{16} d'un microprocesseur de 8 bits. Notez que chaque mot a une longueur d'un octet

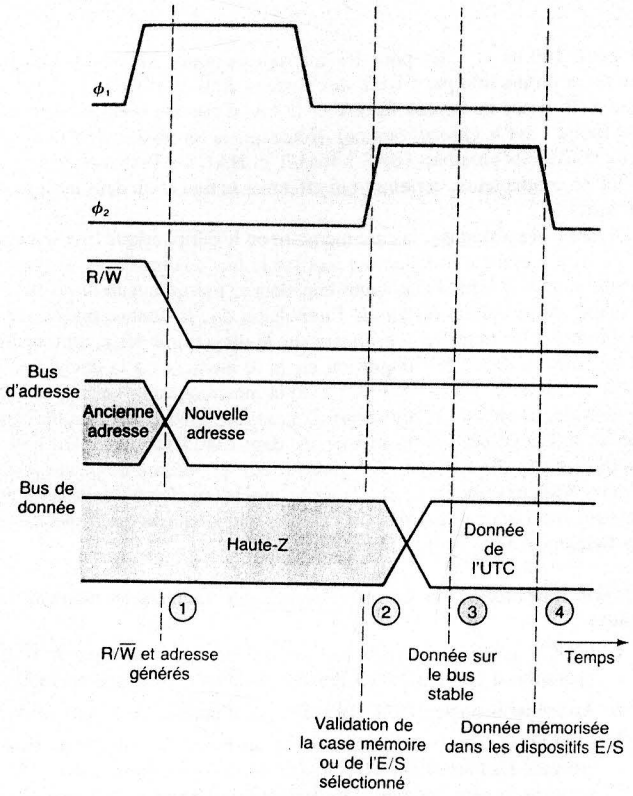


FIGURE 12-11 Synchronisation caractéristique d'une opération d'écriture.

(8 bits). Il faut deux octets successifs pour représenter le code d'adresse 16 bits de l'opérande. Déterminez le nombre total d'opérations de lecture et d'écriture que l'UCT effectue durant l'exécution de ce programme.

Adresse mémoire (Hex)	Mot en mémoire (Hex)	Mnémonique	Description
0020	5B	LDA$5001	Charge accumulateur (ACC) avec X
0021	01		
0022	50		[Adresse de l'opérande X]
0023	6D	ADD$5002	Additionne Y au contenu de ACC
0024	02		
0025	50		[Adresse de l'opérande Y]
0026	8A	STA$5003	Stocke le contenu de ACC
0027	03		
0028	50		[Adresse où ACC doit être mémorisé]
0029	3F	HLT	Fin du programme

Solution L'UCT commence l'exécution du programme en lisant le contenu de la case mémoire 0020. Le mot qui s'y trouve (5B) est amené dans cette unité pour y être décodé comme code op d'une instruction. (Autrement dit, l'UCT tient *toujours* pour acquis que le premier mot d'un programme est un code d'opération et le programme doit *toujours* se conformer à ce format.) L'unité décode ce code op en vue de déterminer l'opération à effectuer et l'adresse de l'opérande associée à cette instruction. Dans la première instruction, une adresse d'opérande est attendue et le programme la trouve dans les deux octets suivants du programme. Ainsi, l'unité lit les emplacements 0021 et 0022 afin d'obtenir l'adresse de la donnée X. Une fois que cette adresse a été lue dans l'UCT, cette dernière poursuit l'exécution de l'instruction LDA en allant chercher le mot dans l'adresse 5001 et en le chargeant dans l'accumulateur. Ceci met un terme à la première instruction et a exigé quatre opérations de lecture distinctes, une pour le code op, deux pour l'adresse et une pour la fonction LDA.

De la même façon, quatre opérations de lecture sont nécessaires pour la seconde instruction qui début en 0023. Cette instruction est aussi associée à une adresse d'opérande de deux octets; il faut donc lire la donnée référencée par cette adresse et la transférer dans l'unité arithmétique de l'UCT.

Le code op de la troisième instruction, rangé dans 0026, nécessite une adresse d'opérande de 2 octets. L'UCT a besoin de *trois* autres lectures pour aboutir jusque là. Cependant, l'instruction STA commande une opération d'écriture quand l'UCT transfère le contenu de l'accumulateur dans l'emplacement mémoire 5003.

La dernière instruction, en 0029, est simplement un code op sans adresse d'opérande. L'unité lit ce code (3F) et le décode, ensuite il met un terme aux opérations, comme on vient de le lui demander.

Le nombre total de lectures est *douze* et le nombre d'écritures, *une*.

QUESTIONS DE RÉVISION

1. Décrivez les étapes qui se déroulent quand une UCT réalise une opération de lecture.
2. Même question, mais cette fois-ci pour une opération d'écriture.
3. Quel élément place une donnée sur le bus de donnée pendant une opération d'écriture?

12-11 CONCLUSION

Pour des raisons de nature pratique, nous avons dû nous limiter à une simple introduction des principes de base, de la terminologie et du fonctionnement des micro-ordinateurs et des microprocesseurs. Nous n'avons même pas aborder les étonnantes capacités et possibilités de ces appareils. De nos jours, toutes les applications techniques tirent avantage de la commande informatisée que rendent possible les microprocesseurs., notamment les micro-ondes, la commande des feux de circulation, les ordinateurs personnels, les appareils de mesure électronique, la régulation de procédés industriels, les jeux électroniques, le contrôle des gaz d'échappement des automobiles et une foule sans cesse grandissante de nouveaux produits.

Le revirement que vient de faire subir à l'industrie de l'électronique le microprocesseur nous porte à croire que d'ici peu toute personne oeuvrant en électronique devra connaître parfaitement le fonctionnement de ces éléments. La présente introduction aura permis, nous l'espérons, de bien asseoir vos connaissances en vue d'études plus poussées.

RÉPONSES AUX QUESTIONS DE RÉVISION

SECTION 12-4

1. Entrée, sortie, arithmétique-logique, commande, mémoire. Voyez texte pour les fonctions. **2.** La combinaison de l'unité arithmétique et logique et de l'unité de commande. **3.** La synchronisation de la transmission d'information numérique entre l'ordinateur et les périphériques E/S.

SECTION 12-5

1. Le microprocesseur (MP) est la partie de traitement centrale du micro-ordinateur.
2. MP, VIVE, MORTE, entrée, sortie.
3. Synchronisation et commande, registre et UAL.

SECTION 12-6

1. Données et instructions. **2.** Exécute les programmes plus rapidement.

SECTION 12-7

1. Un code binaire qui représente l'opération qu'on demande à l'ordinateur d'effectuer.
2. L'adresse de la donnée sur laquelle l'UCT agit pendant l'exécution de l'opération exigée par le code op. **3.** Code op.

SECTION 12-8

1. Un programme en langage machine comprend des codes d'instruction en binaire conservés dans la mémoire. Un programme rédigé dans un langage évolué doit être converti en langage machine avant de pouvoir être exécuté. **2.** Une forme abrégée de l'opération. **3.** Durant un cycle de prise en charge, l'UCT retrouve en mémoire le code op et l'adresse de l'opérande. Durant un cycle d'exécution, l'UCT exécute l'opération commandée par le code op. Le PI conserve l'information sur les adresses des instructions en mémoire.

SECTION 12-9

1. Bus de donnée — transporte des données entre l'UCT et la mémoire et les dispositifs d'entrée/sortie; le bus d'adresse — véhicule le code d'adresse de l'UCT vers la mémoire ou les éléments E/S; le bus de commande — transporte les signaux de cadencement et de synchronisation.

SECTION 12-10

1. Voyez texte. **2.** Voyez texte.
3. L'UCT.

APPENDICE I GLOSSAIRE

Accumulateur Registre principal de l'unité arithmétique et logique (UAL).

ACL Affichage à cristaux liquides.

Acquisition de données Processus suivant lequel un ordinateur recueille des données analogiques numérisées.

Actionneur Élément commandé électriquement qui règle une grandeur physique.

Additionneur complet Circuit logique possédant trois entrées et deux sorties. Les entrées sont un bit de report (C_{EN}) délivré par l'étage précédent, un bit provenant du cumulande et un bit provenant du cumulateur. Les sorties sont un bit de somme et un bit de report (C_{SORTIE}) résultant de l'addition du bit du cumulande avec le bit du cumulateur et de C_{EN}.

Additionneur DCB Additionneur spécial formé de deux additionneurs 4 bits parallèles et d'un circuit détecteur de la correction. Quand la somme de deux groupes codés DCB dépasse 1001_2 (9_{10}), le circuit de détection de la correction le sait et additionne au résultat le facteur de correction, soit 0110_2 (6_{10}), et génère un report sur le rang décimal suivant.

Adresse Nombre qui désigne de façon exclusive l'emplacement d'un mot en mémoire.

Algèbre de Boole Ensemble de règles spéciales utilisées dans la conception et l'analyse des systèmes numériques. En algèbre de Boole, il n'y a que deux valeurs possibles, 0 et 1.

Autotest au démarrage Programme stocké en mémoire MORTE et exécuté par l'UCT à la mise sous tension; il teste la mémoire VIVE et les parties en mémoire MORTE des circuits de l'ordinateur.

Bascule D déclenchée Type de bascule dans lequel l'entrée *D* est l'entrée de synchronisation.

Bascule de Schmitt Circuit numérique qui accepte en entrée un signal variant lentement et qui produit en sortie une transition rapide, franche et exempte d'oscillations.

Bascule Élément de mémoire capable de conserver u niveau logique.

Bascule J–K déclenchée Type de bascule dont les entré *J* et *K* sont des entrées de synchronisation.

Bascule maître-esclave Bascule désormais désuète q possédait deux bascules , une maître et une esclave.

Bascule S–C déclenchée Type de bascule dont les entré S (RAU) et C (RAZ) sont des entrées de synchronisatio

BCD/DEC Mention figurant à l'intérieur d'un symbo IEEE/ANSI et qui signifie une fonction de conversi DCB-décimale avec détection.

Basculement Processus de passage d'un état binaire l'autre.

Bascules déclenchées Bascules dotées d'une entré d'horloge.

BIN/OCT Mention figurant à l'intérieur d'un symbo IEEE/ANSI et qui signifie une fonction de décodage binair octal.

Bit Contraction de l'expression anglaise Binary digi désignant un chiffre du système binaire, soit 0 ou 1.

Bit de parité Bit supplémentaire associé à chaque group codé de façon à ce que la somme des 1 transmis so toujours un nombre pair (ou impair).

Bit de poids le plus faible (LSB) Le bit le plus à droit dans un nombre binaire ou dont le poids est le plus faible

Bit de signe Bit ajouté à gauche du rang le plus fo d'un nombre binaire et qui indique si le nombre est positi ou négatif.

Bloc de contrôle commun Symbole employé dans l norme IEEE/ANSI pour décrire le fait qu'une ou plusieur entrées sont communes à plus d'un circuit du CI.

Bus d'adresse Faisceau de lignes unidirectionnelles qu véhicule le code d'adresse de l'UCT vers la mémoire e les dispositifs E/S.

Bus de commande Faisceau de conducteurs utilisé pour synchroniser les actions de l'UCT et pour isoler les éléments du microprocesseur.

Bus de donnée Conducteurs bidirectionnels qui transportent une donnée entre l'UCT et la mémoire ou un dispositif E/S.

C Signe figurant à l'intérieur d'un symbole IEEE/ANSI et qui signifie que les entrées contrôlent la pénétration de la donnée dans un élément de mémoire.

CAN à approximations successives Type de convertisseur analogique-numérique dans lequel un registre parallèle interne et des circuits logiques complexes servent à réaliser la conversion. La durée de conversion est toujours la même quelle que soit la valeur de l'entrée analogique.

CAN parallèle Type de convertisseur analogique-numérique le plus rapide de tous.

CAN rampe numérique bidirectionnel Type de convertisseur analogique-numérique qui, en recourant à un compteur réversible, augmente ou diminue la tension fournie par un CNA jusqu'à ce que cette tension coupe l'entrée analogique.

CAN rampe numérique Type de convertisseur analogique-numérique dans lequel une forme d'onde interne en escalier est générée et utilisée dans le processus de la conversion. La durée de conversion pour ce type de convertisseur varie selon la valeur du signal analogique.

CAN tension-fréquence Type de convertisseur analogique-numérique qui transforme la tension analogique en une fréquence qu'un compteur dénombre afin de produire une signal numérique.

Capacité Quantité d'emplacements mémoires exprimée en nombre de bits ou nombre de mots.

Cellule mémoire Élément de mémoire qui ne retient qu'un bit.

Charge unitaire (UL) Unité développée par certains fabricants pour spécifier les courants d'entrée et de sortie d'un dispositif. Dans un circuit TTL standard, 1 UL dans l'état HAUT correspond à 40 μA, tandis que dans l'état BAS, cette unité vaut 1,6 mA.

Chiffre de poids le plus faible (LSD) Dans un nombre donné, le chiffre de rang inférieur ou celui dont le poids est le plus faible.

Chronogramme Représentation des niveaux logiques en fonction du temps.

CI bipolaire Circuit numérique intégré ayant comme principaux constituants des transistors NPN et PNP.

CI unipolaires Circuits numériques intégrés dans lesquels des transistors à effet de champ MOS sont les principaux éléments constitutifs.

Circuit échantillonnage-mémorisation Type de circuit dans lequel figurent un tampon à gain unité et un condensateur; il permet de stabiliser le processus de conversion analogique-numérique.

Circuit logique Tout circuit dont la conduite s'explique par un ensemble de règles logiques.

Circuit NI exclusif Circuit à deux entrées dont la sortie devient HAUTE seulement quand les deux entrées sont identiques.

Circuit OU exclusif Circuit à deux entrées dont la sortie devient HAUTE seulement quand les deux entrées sont différentes.

Circuits de blocage Circuits logiques qui régissent le passage d'un signal d'entrée vers la sortie.

Circuits intégrés numériques Circuits numériques sous boîtier réalisés au moyen d'une des nombreuses technologies de fabrication des circuits intégrés.

Circuits logiques combinatoires Circuits constitués de combinaisons de portes logiques; dans ces combinaisons, il n'y a pas de rétroaction de sorties sur des entrées.

CMOS (Complementary Metal-Oxide-Semiconductor) Technologie des circuits intégrés dont le composant de base est le transistor MOS. Cette famille de circuits logiques appartient à la catégorie des CI numériques bipolaires.

CNA à réseau R/2R Type de convertisseur numérique-analogique dont les résistances sont dans un rapport 2 pour 1.

CNA bipolaire Convertisseur numérique-analogique dont l'entrée est un nombre binaire signé et la sortie est le signal analogique équivalent (négatif ou positif).

Codage binaire pur Représentation d'un nombre décimal au moyen de son équivalent binaire pur.

Codage Processus de mise en correspondance entre un groupe de symboles et des chiffres, des lettres ou des mots.

Code ASCII (American Standard Code for Information Interchange) Code alphanumérique à six éléments employé par la plupart des constructeurs d'ordinateurs.

Code décimal codé binaire (DCB) Un code sur 4 bits dans lequel chaque chiffre décimal est représenté par son équivalent binaire de 4 bits.

Code Gray Code dans lequel jamais plus d'un bit n'est modifié entre un élément de ce code et l'élément suivant.

Code majoré de 3 Code de quatre bits dans lequel on additionne 3 à chaque chiffre décimal avant son codage en binaire.

Codes alphanumériques Codes constitués pour représenter des nombres, des lettres, des signes de ponctuation et des caractères spéciaux.

Codeur Circuit numérique qui fournit en sortie un code correspondant aux entrées activées.

Codeur de priorité Type de codeur spécial qui peut détecter le moment où deux entrées sont actives en même temps et qui sort ensuite le code de l'entrée ayant le nombre le plus élevé.

Codeur un parmi huit Circuit numérique qui produit un code de 3 bits correspondant à l'entrée activée parmi les 8 possibles.

Comparateur de grandeurs Circuit numérique qui compare deux entrées binaires et produit en sortie un signal qui indique si ces grandeurs sont égales, différentes, et dans ce dernier cas, quelle grandeur est la plus grande.

Complément Voyez inversion.

Compteur à décade Tout compteur pouvant occuper dix états logiques différents.

Compteur à propagation Voyez Compteur asynchrone.

Compteur asynchrone Type de compteur dans lequel chaque sortie de bascule devient l'horloge d'entrée de la bascule suivante dans la chaîne.

Compteur binaire Groupe de bascules raccordées de façon spéciale et dont les états constituent l'expression en binaire du nombre d'impulsions appliquées à l'entrée du compteur.

Compteur Compteur qui dénombre progressivement de zéro à une valeur maximale.

Compteur DCB Compteur binaire qui dénombre les valeurs de 0000_2 à 1001_2 avant de recommencer son cycle.

Compteur en anneau Registre à décalage dont la sortie de la dernière bascule est rétrocouplée à l'entrée de la première bascule.

Compteur Johnson Registre à décalage dont la sortie complémentée est ramenée à l'entrée de la première bascule.

Compteur préréglable Compteur que l'on peut initialiser avec un nombre quelconque, soit de façon synchrone ou de façon asynchrone.

Compteur synchrone Compteur dans lequel toutes les bascules sont déclenchées en même temps.

Conflit de bus Situation qui se produit quand les sorties d'au moins deux dispositifs actifs sont placées sur le bus en même temps.

Contrôleur de parité Circuit qui saisit un ensemble de bits (y compris le bit de parité) et vérifie que la parité est la bonne.

Contrôleur DRAM CI servant à gérer le rafraîchissement et le multiplexage, opérations indispensables dans les DRAM (mémoires vives dynamiques).

Conversion parallèle-série Procédure suivant laquelle des données fournies simultanément sont restituées en sortie un bit à la fois.

Convertisseur numérique–analogique (CNA) Circuit qui convertit une entrée numérique en un signal analogique correspondant.

Convertisseur analogique-numérique double rampe Type de convertisseur analogique-numérique qui charge linéairement un condensateur au moyen d'un courant proportionnel à V_A pendant un intervalle fixe; ensuite il incrémente un compteur à mesure que ce condensateur se décharge linéairement jusqu'à zéro.

Convertisseur analogique-numérique (CAN) Circuit qui transforme un signal analogique en un signal de sortie numérique.

CT = 0 Libellé employé pour qualifier une entrée dans un symbole IEEE/ANSI; il signifie que le compteur repart à zéro quand l'entrée passe à son niveau vrai.

CTR Libellé employé à l'intérieur d'un symbole IEEE/ANSI pour indiquer que le CI est un compteur.

Cumulande Nombre auquel on additionne un nombre.

Cumulateur Nombre ajouté à un autre.

Cycle d'exécution Période pendant laquelle l'unité de commande de l'ordinateur effectue l'opération désignée par le code op récupéré.

Cycle de prise en charge Période durant laquelle on récupère un code d'instruction gardé en mémoire.

D Libellé dans un symbole IEEE/ANSI signifiant « donnée ».

DBS Diode à barrière de Schottky que l'on retrouve dans toutes les séries TTL Schottky.

Déclenché par un front Désignation de la façon selon laquelle la bascule est commandée par une transition. Une bascule peut être déclenchée par un front montant ou par un front descendant.

Décodeur 4 par 10 Voyez Décodeur DCB-décimal.

Décodeur Circuit numérique qui convertit un code d'entrée binaire en un seul autre code numérique correspondant.

Décodeur DCB-décimal Décodeur qui convertit une entrée DCB en son équivalent décimal.

Décodeur/pilote DCB-7 segments Circuit numérique dont l'entrée est une valeur DCB et la sortie, les signaux nécessaires pour activer les segments d'un afficheur à 7 éléments qui réalisent le chiffre décimal équivalent.

Décompteur Compteur qui dénombre d'une valeur maximale jusqu'à zéro.

Découplage de l'alimentation Raccordement d'un petit condensateur RF entre la masse et V_{CC} à proximité de chaque circuit intégré TTL sur une plaquette de circuits.

DEL Diode électroluminescente.

Demi-additionneur Circuit logique possédant deux entrées et deux sorties. Les entrées sont un bit pour le cumulande et un bit pour le cumulateur, respectivement. Les sorties sont un bit de somme résultant de l'addition du bit du cumulande et du bit du cumulateur, et un bit de report (C_{SORTIE}) ramené à l'étage suivant.

Démultiplexeur (DEMUX) Circuit logique qui selon l'état de ses entrées de sélection achemine les données entrées vers une sortie parmi plusieurs.

Dérive d'horloge Arrivée d'un signal d'horloge aux bornes d'horloge de différentes bascules à des instants différents en raison des retards de propagation.

Détecteur de front Circuit qui produit, au même instant que le front actif d'un signal de déclenchement, une petite pointe de tension.

Diagramme de Karnaugh Diagramme bidimensionnel qui réécrit la table de vérité et que l'on utilise pour simplifier une somme de produits.

Diminuande Nombre duquel on retranche le diminuteur.

Diminuteur Nombre que l'on soustrait du diminuande.

DIP Boîtier à double rangée de broches. Le type de boîtier le plus répandu.

Élément logique programmable (PLD) CI qui renferme un grand nombre de fonctions logiques déjà connectées. L'utilisateur peut programmer une certaine fonction dans ce CI en brisant sélectivement les interconnexions appropriées.

Disque dur Support magnétique en métal servant de mémoire de masse.

Disquette Support magnétique souple servant de mémoire de masse.

Distributeur de données Voyez Démultiplexeur.

Durées de transition de l'horloge Durée de montée et de descente des transitions du signal d'horloge dans une certaine puce, paramètre indiqué par le fabricant.

ECL Signifie Emitter-coupled logic, également appelé *logique à mode courant*.

Écrasement de tête Contact entre la tête de lecture/écriture d'un disque dur et la surface, qui peut détériorer des données.

Écriture Expression employée pour décrire la condition qui existe quand l'UCT envoie une donnée à un autre élément.

EEPROM (mémoire MORTE programmable et effaçable électroniquement) Mémoire MORTE que l'on peut programmer, effacer et reprogrammer par voie électrique.

Élément de mémoire D Circuit formé d'un élément de mémoire à portes NON-ET et deux portes ET d'aiguillage.

Élément de mémoire Un type de bascule.

Élément PAL Classe d'éléments logiques programmables. Son réseau ET est programmable, tandis que son réseau OU est câblé.

Élément PLA Classe d'éléments logiques programmables. Ses réseaux, autant ET que OU, sont programmables. Désigné également par FPLA pour *field-programmable logic array*.

Entrée flottante Borne d'entrée non raccordée dans un circuit logique.

Entrées asynchrones Entrées d'une bascule qui peuvent changer la réaction d'une bascule indépendamment des entrées synchrones et d'horloge.

Entrées de commande Signaux d'entrée synchronisés sur la transition d'horloge active et qui commandent la valeur de l'état de sortie de la bascule.

Entrées synchrones Voyez Entrées de commande.

EPROM (mémoire MORTE programmable et effaçable) Mémoire MORTE programmable électriquement par l'utilisateur et que l'on peut effacer (avec des rayonnements ultraviolets) et reprogrammer autant de fois qu'on le veut.

Erreur de linéarité Expression employée par certains fabricants de convertisseurs numériques-analogiques pour caractériser l'exactitude de l'appareil. C'est l'écart maximal du pas de progression par rapport au pas de progression idéal.

Erreur de quantification Erreur imputable au fait que la résolution n'est pas nulle dans un convertisseur analogique-numérique. Cette erreur est inhérente à l'appareil.

Erreur pleine échelle Expression employée par certains fabricants de convertisseurs pour spécifier l'exactitude d'un convertisseur numérique-analogique. Elle désigne l'écart maximal de la sortie du convertisseur par rapport à la valeur vraie attendue.

ET câblé Expression utilisée pour décrire la fonction logique obtenue quand on relie des sorties en collecteur ouvert.

ET& Signe figurant à l'intérieur d'un symbole IEEE/ANSI et qui signifie une porte ET ou une fonction de multiplication logique (ET).

État quasi-stable État qu'occupe temporairement un monostable déclenché (habituellement Q = 1) avant de revenir dans son état stable (habituellement $Q = 0$).

État RAZ État d'une bascule quand Q = 0.

État Un L'état $Q = 1$ d'une bascule.

Facteur de charge Voyez Sortance.

Forme d'onde en escalier Type de forme d'onde que l'on retrouve en sortie d'un convertisseur numérique-analogique quand l'entrée numérique est incrémentée régulièrement.

Forme en complément à 2 Résultat de l'addition d'un 1 au bit de poids le plus faible d'un nombre binaire exprimé dans sa forme en complément à 1.

Forme en somme de produits Expression logique formée de deux ou plusieurs termes ET (produits) réunis par une porte OU (somme).

Fréquence de déclenchement maximale (f_{MAX}) Fréquence la plus élevée que peut posséder le signal d'horloge et qui continue de déclencher correctement la bascule.

Fréquencemètre Circuit qui mesure et affiche la fréquence d'un signal.

Front montant (FM) Dans un signal d'horloge, la transition entre 0 et 1.

G Libellé employé en rapport avec une entrée dans un symbole IEEE/ANSI afin d'indiquer une dépendance fonctionnelle de type ET.

Générateur de fonctions Circuit qui produit plusieurs types de formes d'ondes; renferme souvent de la mémoire VIVE, un CNA et un compteur.

Générateur de parité Circuit qui saisie un ensemble de bits et produit le bit de parité approprié pour la donnée.

Génération de fonctions logiques Réalisation d'une fonction logique directement à partir de la table de vérité en utilisant un CI numérique comme un multiplexeur.

Grand ordinateur Voyez Superordinateur.

Horloge Signal numérique qui se présente comme un train d'impulsions rectangulaires ou carrées.

HPRI/BCD Libellé utilisé à l'intérieur d'un symbole IEEE/ANSI pour indiquer que la fonction du CI est de convertir l'entrée de donnée ayant la priorité en son code DCB correspondant.

IEEE/ANSI Institute of Electrical and Electronics Engineers/American National Standards Institute.

Indéterminé Se dit d'un niveau de tension qui est situé hors des plages de tensions qui définissent le 0 et le 1 logique.

Indifférent Qualificatif donné à un niveau de sortie d'un circuit quand, pour les conditions d'entrée en cours, il peut être aussi bien à 0 qu'à 1.

Interfaçage Réunion de dispositifs différents afin qu'ils puissent travailler de concert et en toute compatibilité; connexion de la sortie d'un système à l'entrée d'un système ayant des caractéristiques électriques différentes.

Interrupteur bilatéral Circuit CMOS se comportant comme un interrupteur unipolaire et undirectionnel contrôlé par un niveau logique.

Intervalle d'échantillonnage Plage de temps durant laquelle un fréquencemètre échantillonne un signal et détermine sa fréquence.

Inverseur Aussi appelé circuit NON; circuit logique qui matérialise la fonction NON. Un INVERSEUR possède une entrée et une sortie qui est toujours le complément de l'entrée.

Inversion Basculement d'un niveau logique dans son état contraire.

Langage évolué Langage de programmation informatique qui utilise des mots de la langue anglaise pour désigner des instructions afin de faciliter la rédaction des programmes.

Langage machine Langage de programmation informatique dont les instructions sont représentées par des groupes de 0 et de 1. Le langage machine est le seul langage compris par l'ordinateur.

Lecture Expression employée pour indiquer que l'UCT reçoit une donnée d'un autre élément.

Ligne de données bidirectionnelle Expression attribuée à une ligne de données qui sert autant de ligne d'entrée que de ligne de sortie selon l'état des entrées de validation.

Logique à consommation de courant Famille des dispositifs logiques dont les sorties consomment du courant fourni par les entrées qu'elles pilotent.

Logique à fourniture de courant Famille de dispositifs logiques dont les sorties fournissent du courant aux entrées qu'elles pilotent.

LSI Intégration à grande échelle (100 à 9999 portes).

Mémoire à accès sélectif Mémoire dans laquelle le temps d'accès est le même pour tous les emplacements.

Mémoire à accès séquentiel Mémoire dont le temps d'accès varie selon l'endroit où se trouve la donnée à retrouver.

Mémoire à bande magnétique Mémoire de masse qui conserve les données dans de petits points aimantés sur une bande de plastique recouverte d'une couche magnétique.

Mémoire à bulles magnétiques (MBM) Dispositif de stockage de masse, à semiconducteurs, rémanent, séquentiel et comprenant des minuscules domaines magnétiques (bulles).

Mémoire à disque magnétique Mémoire de masse qui conserve les données dans de petits points aimantés sur la surface d'un disque tournant.

Mémoire à tores Mémoire morte permanente formée de petits tores métalliques.

Mémoire Aptitude propre à un circuit qui lui permet de conserver un état même après que la condition d'entrée à l'origine de cet état a disparue.

Mémoire de masse Organe de stockage de grands volumes de données. N'est pas une partie de la mémoire interne d'un ordinateur.

Mémoire FIFO (Premier entrée, premier sortie) Mémoire morte à semiconducteurs dont les mots sont lus dans l'ordre de leur écriture.

Mémoire MORTE Mémoire dont le rapport des lectures sur celui des écritures est très grand.

Mémoire VIVE dynamique (DRAM) Type de mémoire à semiconducteurs qui stocke les données sous forme de charges de condensateur et qu'il faut donc rafraîchir périodiquement.

Mémoire VIVE Mémoire dans laquelle on effectue presque autant d'écritures que de lectures.

Mémoire VIVE statique (SRAM) Mémoire VIVE à semiconducteurs qui stocke l'information dans des bascules et qu'il n'est pas nécessaire de rafraîchir périodiquement.

Mémoire volatile Mémoire qui doit rester sous tension pour conserver les données qu'on a placées en elle.

Méthode de parité Solution de détection des erreurs mise en oeuvre durant une transmission de données.

Micro-ordinateur Dernier-né de la famille des ordinateurs; il est formé d'une puce microprocesseur, de puces d'interface E/S. Dans certains cas, tous les circuits du micro-ordinateur sont dans une seule puce.

Microprocesseur Puce LSI qui renferme l'unité centrale de traitement (UCT).

Miniordinateur Machine généralement plus grande et plus rapide qu'un micro-ordinateur.

Minuterie 555 CI compatible TTL que l'on peut câbler suivant différents modes pour réaliser un monostable ou un multivibrateur astable.

Monostable redéclenchable Type de monostable qui réagit à un signal de déclenchement même quand il se trouve dans son état quasi-stable.

Mot machine Groupe de bits qui constitue l'élément d'information premier dans un ordinateur.

Mot mémoire Groupe de bits en mémoire qui correspond à une instruction ou à une donnée.

MROM (mémoire MORTE programmée par masque) Mémoire MORTE programmée par le fabricant selon les spécifications du client. Elle ne peut être ni effacée ni reprogrammée.

Multiplexage d'adresse Multiplexage intervenant dans les mémoires VIVES dynamiques, dont la raison d'être est d'utiliser moins de broches; les deux moitiés de l'adresse sont mémorisées par des séquences séparées dans le circuit.

Multiplicateur binaire Circuit numérique spécial en mesure d'effectuer l'opération de multiplication de deux nombres binaires.

Multivibrateur astable Circuit numérique qui oscille entre deux états de sortie instables.

Multivibrateur bistable Nom donné parfois à une bascule.

Niveau logique État d'une variable de tension. L'état 1 (HAUT) et l'état 0 (BAS) correspondent aux deux plages de tensions définies pour un dispositif numérique.

Niveau logique vrai Niveau logique d'un circuit considéré comme vrai. Si le symbole du circuit renferme un rond, ce circuit est dit vrai au niveau BAS. Au contraire, sans rond, le circuit est dit vrai au niveau HAUT.

Non validé Expression employée pour dire qu'un signal logique n'a pas d'effet, synonyme « d'inactif ».

Notation fonctionnelle Méthode de représentation symbolique de la relation entre les entrées et les sorties des circuits logiques. Cette méthode a recours à des symboles significatifs placés près du centre et en haut du symbole ou en son centre géométrique.

Numérisation Processus de conversion d'un signal analogique en signal numérique.

Octet Mot de 8 bits.

Opération d'écriture Opération consistant à placer un nouveau mot dans une case mémoire précise.

Opération de lecture Opération au cours de laquelle un mot dans une case mémoire précise est trouvé et souvent transféré à un autre dispositif.

Opération ET Opération de l'algèbre booléenne dans laquelle le symbole désigne une multiplication logique (ET) de deux variables logiques ou plus. Le résultat d'une opération logique est un niveau HAUT si chaque entrée est aussi à un niveau HAUT.

Ordinateur numérique Système matériel qui effectue des opérations arithmétiques et logiques, manipule des données et prend des décisions.

Oscillateur commandé par tension (VCO) Circuit qui produit un signal de sortie dont la fréquence est proportionnelle à la tension appliquée en entrée.

Oscilloscope numérique à mémoire Appareil de mesure qui échantillonne, numérise, mémorise et affiche des tensions analogiques.

Pas de progression Voyez Résolution.

Pilote Terme technique parfois ajouté à la description d'un CI en vue d'indiquer que les sorties de ce module supportent des niveaux de courant et de tension supérieurs aux valeurs normalisées pour ce CI.

Pilotes de bus Circuits qui retiennent temporairement les sorties de dispositifs raccordés à un bus commun; utilisés quand un grand nombre de dispositifs partagent le même bus.

Plan arrière Connexion électrique commune de tous les segments d'un afficheur à DEL.

Pointeur d'instruction Registre de l'UCT qui conserve l'adresse de l'instruction suivante devant être prise en charge.

Porte de transmission Voyez Interrupteur bilatéral.

Porte ET Porte d'un circuit numérique qui matérialise l'opération ET. La sortie d'une telle porte est HAUTE (niveau logique 1) seulement quand toutes ses entrées sont au niveau HAUT.

PRESET Entrée asynchrone utilisée pour mettre immédiatement $Q = 1$.

Produit vitesse-consommation Valeur chiffrée (en joules) sur laquelle on se base souvent pour apprécier différentes familles logiques. On l'obtient en multipliant le retard de propagation par la consommation d'un circuit logique.

Programme d'amorçage Programme stocké en mémoire MORTE qu'un ordinateur exécute au moment de sa mise en route.

Programme Séquence d'instructions binaires organisée de manière à faire accomplir une certaine tâche à l'ordinateur.

PROM (Mémoire MORTE programmable) Mémoire morte que l'utilisateur peut programmer par voie électrique. Par la suite, elle n'est plus effaçable ni reprogrammable.

Propagation du report Retard intrinsèque aux circuits de certains additionneurs parallèles qui empêche le bit de report (C_{sort}) et le résultat de l'addition de sortir en même temps du circuit.

R Libellé utilisé dans un symbole IEEE/ANSI pour signaler une fonction de remise à zéro.

Rafraîchissement Processus de recharge des cellules dans une mémoire dynamique.

RAU CC Entrée asynchrone d'une bascule qui sert à mettre Q immédiatement à 1.

RAZ CC Entrée asynchrone d'une bascule qui sert à mettre Q immédiatement à zéro.

RAZ Entrée asynchrone d'une bascule qui sert à mettre Q immédiatement à 0.

Registre à décalage Circuit numérique qui accepte d'une source des données binaires qu'il décale dans sa chaîne de bascules un bit à la fois.

Registre à décalage en anneau Registre à décalage dont les sorties de la dernière bascule sont ramenées aux entrées de la première.

Registre à écriture parallèle/lecture série Type de registre dans lequel la donnée est placée en parallèle et est lue en série sur la seule broche de sortie.

Registre à écriture parallèle/lecture parallèle Type de registre dans lequel la donnée est chargée en parallèle et qui possède un groupe de sorties parallèles.

Registre à écriture série/lecture parallèle Type de registre dans lequel les données arrivent en série et sont communiquées d'un bloc sur des sorties parallèles.

Registre à écriture série/lecture série Type de registre dans lequel les données arrivent en série et sont communiquées également en série sur la seule borne de sortie.

Registre Groupe de bascules pouvant mémoriser une donnée.

Registre tampon Registre qui retient temporairement des données numériques.

Report anticipé Capacité qu'ont certains additionneurs parallèles de prévoir s'il y a aura un bit de report ou non (C_{SORTIE}) à la suite de l'addition, et cela sans attendre la propagation des bits dans tous les étages; cette capacité réduit les retards de propagation.

Report Bit ou chiffre généré quand la somme de deux chiffres est supérieure à la base du système de numération avec lequel on travaille.

Représentation analogique Représentation d'une grandeur qui prend toutes les valeurs d'un intervalle continu.

Représentation numérique Représentation d'une grandeur qui varie par pas discrets à l'intérieur d'une plage.

RESET Terme synonyme de « RAZ ». L'état Q = 0 d'une bascule.

Résolution Dans un convertisseur numérique-analogique, c'est la plus petite variation de la sortie qui correspond à un changement de l'entrée numérique. Appelée aussi *pas de progression*. Dans un convertisseur analogique-numérique, c'est la plus petite variation de l'entrée analogique qui correspond à un changement sur la sortie numérique.

Résolution en pourcentage Dans un convertisseur numérique-analogique, rapport du pas de progression sur la pleine échelle. Aussi, l'inverse du nombre maximal de pas pour ce convertisseur.

Retard de propagation (t_{PLH}/t_{PHL}) Durée entre l'instant où on applique un signal et l'instant où la sortie réagit.

Réunion Action de combiner des cases adjacentes dans un diagramme de Karnaugh contenant des 1, dans un but de simplification de l'expression somme de produits.

Ronds Petits ronds placés aux entrées ou aux sorties de symboles de circuits logiques qui indiquent l'inversion d'un signal particulier. La présence d'un rond indique que l'entrée ou la sortie est vraie au niveau BAS.

Sélecteur de données Voyez Multiplexeur.

SET Terme synonyme de RAU CC.

Sigma Σ Lettre grecque symbolisant une addition; on l'utilise fréquemment pour signifier un bit de somme dans un additionneur.

Somme de contrôle Mot spécial mémorisé en mémoire MORTE. Cette somme découle de l'addition de tous les autres mots dans la mémoire MORTE et sert à des fins de contrôle des erreurs.

Sonde logique Outil de dépannage numérique qui décèle et indique le niveau logique à un point précis du circuit.

Sortance Nombre maximal d'entrées logiques standard que la sortie d'un circuit numérique peut piloter sans crainte.

Sortie en totem Expression employée pour décrire le montage de deux transistors bipolaires dans lequel est prélevée la sortie de la plupart des circuits TTL.

Sortie pleine échelle Sortie maximale réalisable par un convertisseur numérique-analogique.

SRG_n Libellé que l'on retrouve dans le bloc de contrôle commun d'un symbole IEEE/ANSI et qui signifie que ce CI est un registre à décalage de n bits.

SSI Intégration à petite échelle (moins de 12 portes).

Strobage Technique fréquemment employée pour éliminer les pointes de décodage.

Strobe de l'adresse de colonne (SAC) Signal envoyé pour verrouiller l'adresse de la colonne dans une puce DRAM (mémoire VIVE dynamique).

Strobe de l'adresse de ligne Signal envoyé pour mémoriser l'adresse de ligne dans une puce DRAM.

Substrat Pièce de matériau monocrystallin qui entre dans la fabrication de tous les CI.

Superordinateur Grand système informatique que l'on utilise pour tenir à jour et gérer d'imposantes quantités de données.

Surtension Variation de tension temporaire, pointue et parasite.

Système à poids positionnels Système de numération dont la valeur d'un chiffre est fonction de son rang dans le nombre.

Système analogique Combinaison de dispositifs visant à traiter des grandeurs physiques exprimées sous forme analogique.

Système binaire Système de numération dans lequel les deux seuls chiffres possibles sont 0 et 1.

Système de numération hexadécimal Système de numération dont la base est 16; il utilise les dix chiffres de 0 à 9 et les symboles de A à F pour exprimer ses valeurs.

Système décimal Système de numération qui représente les grandeurs au moyen de dix chiffres ou symboles différents.

Système hydrique Système qui exploite à la fois la technique numérique et la technique analogique.

Système logique séquentiel Système logique dans lequel les sorties logiques et l'ordre des opérations sont conditionnés par les conditions d'entrée actuelles et antérieures.

Système numérique Regroupement de dispositifs dont les actions concourent à manipuler des grandeurs physiques exprimées sous forme numérique.

Systèmes synchrones Systèmes dont les sorties de circuits ne changent d'état qu'au moment des transitions de l'horloge.

Table d'excitation d'un circuit Table regroupant pour chaque bascule les transitions d'état PRÉSENTES et SUIVANTES, ainsi que les niveaux J et K nécessaires.

Table d'excitation J–K Tableau illustrant les conditions d'entrée J et K nécessaires pour chaque transition d'état possible d'une bascule J–K.

Table de vérité Associé à un circuit, tableau qui présente pour toutes les combinaisons de niveaux d'entrée les sorties résultantes.

Tampon deux taux Variante des piles FIFO dans laquelle les données sont écrites à un certain rythme et lues à un autre rythme.

Tampon/pilote Circuit conçu pour que sa sortie délivre un courant ou une tension plus grande que celle d'un circuit normal.

Temps d'accès Durée entre le moment où la mémoire reçoit une nouvelle adresse et le moment où la donnée de sortie peut être lue.

Temps d'établissement Durée que prend la sortie d'un convertisseur numérique-analogique pour aller, à un demi-pas de progression près, de zéro à la valeur pleine échelle, quand son entrée passe de 00...00 à 11...11.

Temps de maintien (t_H) Intervalle de temps qui suit immédiatement la transition active de l'horloge et durant lequel l'entrée de contrôle doit rester au niveau approprié.

Temps de stabilisation (t_s) Durée précédant immédiatement la transition active de l'horloge et pendant laquelle l'entrée de commande doit rester au niveau approprié.

Test de précision statique Test au cours duquel une valeur binaire fixe est appliquée à l'entrée d'un convertisseur numérique-analogique, tandis que l'on relève avec précision la sortie analogique. La mesure obtenue doit se situer dans une plage prévue, spécifiée par le fabricant du convertisseur.

Tête de lecture/écriture Électroaimant en fer « doux » que l'on utilise pour lire ou pour écrire des données sur un disque ou une bande magnétique.

Théorèmes de Boole Propositions reliant des variables booléennes qui permettent de simplifier des expressions logiques.

Théorèmes de De Morgan Premièrement, Théorème spécifiant que le complément d'une somme d'éléments (opération OU) est égal au produit (opération ET) des compléments de ces éléments; et deuxièmement, théorème spécifiant que le complément d'un produit d'éléments (opération ET) est égal à la somme des compléments de ces éléments.

Traceur galvanométrique Appareil de test utilisé pour déceler une variation de courant dans un conducteur ou un chemin de conduction d'une plaquette de circuit imprimé.

Transducteur Dispositif qui transforme une variable physique en une variable électrique (c'est le cas d'une cellule photoélectrique, ou d'un thermocouple).

Transfert asynchrone Transfert de données effectué sans le recours à un signal d'horloge.

Transfert de données en parallèle Opération au cours de laquelle la totalité d'un registre est transférée en même temps dans un autre registre.

Transfert de données série Transfert de données qui a lieu un bit à la fois.

Transfert parallèle Voyez Transfert asynchrone.

Transistor à consommation de courant Nom donné au transistor de sortie (Q_4) d'un circuit TTL. Ce transistor conduit quand le niveau de sortie logique est BAS.

Transistor à fourniture de courant Nom donné au transistor de sortie (Q_3) de la plupart des circuits TTL. Ce transistor conduit quand le niveau de sortie logique est HAUT.

Transistor de rappel à la masse Voyez Transistor à consommation de courant.

Transistor de rappel à la source Voyez Transistor à fourniture de courant.

Transitoires de courant Pointes de courant générées par un montage de sorties en totem dans un circuit TTL grand; les deux transistors sont mis en conduction simultanément.

Transmission en parallèle Transfert simultané d'un endroit à un autre de tous les bits d'un nombre binaire.

Transmission en série Transfert d'informations binaires d'un endroit à un autre, un bit à la fois.

Transparent Se dit d'un mode opératoire de l'élément de mémoire D dans lequel la sortie Q suit l'entrée D.

Signal déclencheur Signal d'entrée d'une bascule ou d'un monostable qui provoque des changements d'états selon les conditions des signaux de commande.

Transposeur de niveau de tension Circuit qui prend en entrée un ensemble de niveaux de tension et fournit en sortie un ensemble différent de niveaux de tension.

Trois états Type de montage qui produit trois états de sortie : HAUT, BAS et haute impédance (Haute-Z).

TTL (Transistor-Transistor Logic) Technologie de fabrication des circuits intégrés qui utilise le transistor comme principal élément de circuit.

TTL faible consommation (TTL-L) Sous-famille TTL dans laquelle on retrouve les mêmes circuits que dans les TTL standard, mais dotés de résistances plus grandes.

TTL Schottky Sous-famille TTL comprenant les mêmes circuits que la famille TTL standard excepté la diode à barrière de Schottky (DBS) qui est montée en parallèle entre la base et le collecteur de chaque transistor.

TTL Schtottky faible consommation (TTL-LS) Sous-famille TTL dans laquelle on retrouve les mêmes circuits que dans les TTL Schottky, mais dotés de résistances plus grandes.

ULSI Intégration à ultragrande échelle (plus de 100 000 portes).

Unité arithmétique et logique (UAL) Circuit numérique que l'on retrouve dans les ordinateurs et qui a la charge des opérations arithmétiques et logiques.

Unité centrale de traitement (UCT) Partie de l'ordinateur qui regroupe l'unité arithmétique et logique (UAL) et l'unité de commande.

Unité d'entrée Partie d'un ordinateur au moyen de laquelle on introduit de l'information dans la mémoire et l'UAL de l'ordinateur.

Unité de commande Partie d'un ordinateur qui assure le décodage des instructions du programme et fournit les signaux de cadencement et de commande nécessaires à leur exécution.

Unité de mémoire Partie d'un ordinateur qui stocke les instructions et les données communiquées par l'unité d'entrée, ainsi que les résultats trouvés par l'unité arithmétique et logique.

Validé Qualificatif utilisé pour indiquer l'état d'un signal logique; synonyme de « vrai ».

Virgule binaire Signe qui sépare la partie entière d'un nombre binaire de sa partie fractionnaire.

VLSI Intégration à très grand échelle (10 000 à 99 999 portes).

/ Signe utilisé dans des libellés de sortie de symboles IEEE/ANSI; son rôle est de séparer deux fonctions.

+ Ce signe à l'intérieur d'un symbole IEEE/ANSI et sur une entrée d'horloge est l'indication que le compteur est incrémenté de 1 au moment du déclenchement par l'horloge.

− Ce signe à l'intérieur d'un symbole IEEE/ANSI et sur une entrée d'horloge est l'indication que le compteur est décrémenté de 1 au moment du déclenchement par l'horloge.

→ Quand on retrouve cette flèche dans le libellé d'une entrée d'un symbole IEEE/ANSI, on sait qu'au moment d'une transition active sur cette entrée, un décalage vers la droite est amorcé.

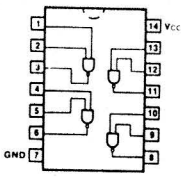
APPENDICE II FICHES TECHNIQUES DES FABRICANTS DE CI

Ces fiches sont reproduites avec l'autorisation de Fairchild, une filiale de Schlumberger Company et de Texas Instruments Incorporated.

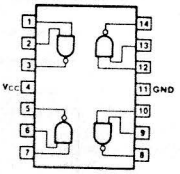
00

54/7400
54H/74H00
54S/74S00
54LS/74LS00
QUAD 2-INPUT NAND GATE

CONNECTION DIAGRAMS
PINOUT A



PINOUT B



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7400PC, 74H00PC 74LS00PC, 74S00PC		9A
Ceramic DIP (D)	A	7400DC, 74H00DC 74LS00DC, 74S00DC	5400DM, 54H00DM 54LS00DM, 54S00DM	6A
Flatpak (F)	A	74LS00FC, 74S00FC	54LS00FM, 54S00FM	3I
	B	7400FC, 74H00FC	5400FM, 54H00FM	

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74H (U.L.) HIGH/LOW	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	1.25/1.25	1.25/1.25	0.5/0.25
Outputs	20/10	12.5/12.5	25/12.5	10/5.0 (2.5)

DC AND AC CHARACTERISTICS: See Section 3*

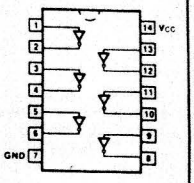
SYMBOL	PARAMETER	54/74 Min	54/74 Max	54/74H Min	54/74H Max	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS	
ICCH	Power Supply	8.0		16.8		16		1.6		mA	VIN = Gnd	VCC = Max
ICCL	Current	22		40		36		4.4			VIN = Open	
tPLH	Propagation Delay	22		10		2.0	4.5	10		ns	Figs. 3-1, 3-4	
tPHL		15		10		2.0	5.0	10				

*DC limits apply over operating temperature range; AC limits apply at TA = +25°C and VCC = +5.0 V.

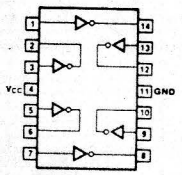
04

54/7404
54H/74H04
54S/74S04
54S/74S04A
54LS/74LS04
HEX INVERTER

CONNECTION DIAGRAMS
PINOUT A



PINOUT B



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7404PC, 74H04PC 74S04PC, 74S04APC 74LS04PC		9A
Ceramic DIP (D)	A	7404DC, 74H04DC 74S04DC, 74S04ADC 74LS04DC	5404DM, 54H04DM 54S04DM, 54S04ADM 54LS04DM	6A
Flatpak (F)	A	74S04FC, 74S04AFC 74LS04FC	54S04FM, 54S04AFM 54LS04FM	3I
	B	7404FC, 74H04FC	5404FM, 54H04FM	

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74H (U.L.) HIGH/LOW	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	1.25/1.25	1.25/1.25	0.5/0.25
Outputs	20/10	12.5/12.5	25/12.5	10/5.0 (2.5)

DC AND AC CHARACTERISTICS: See Section 3*

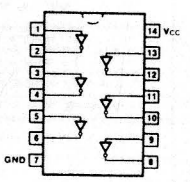
SYMBOL	PARAMETER	54/74 Min	54/74 Max	54/74H Min	54/74H Max	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS	
ICCH	Power Supply	12		26		24		2.4		mA	VIN = Gnd	VCC = Max
ICCL	Current	33		58		54		6.6			VIN = Open	
tPLH	Propagation Delay	22		10		2.0	4.5	10		ns	Fig. 3-1, 3-4	
tPHL		15		10		2.0	5.0	10				
tPLH	Propagation Delay (54/74S04A only)					1.0	3.5			ns	Fig. 3-1, 3-4	
tPHL						1.0	4.0					

*DC limits apply over operating temperature range; AC limits apply at TA = +25°C and VCC = +5.0 V.

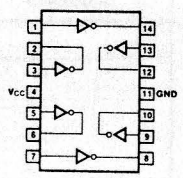
05

54/7405
54H/74H05
54S/74S05
54S/74S05A
54LS/74LS05
HEX INVERTER
(With Open-Collector Output)

CONNECTION DIAGRAMS
PINOUT A



PINOUT B



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V, ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7405PC, 74H05PC 74S05PC, 74S05APC 74LS05PC		9A
Ceramic DIP (D)	A	7405DC, 74H05DC 74S05DC, 74S05ADC 74LS05DC	5405DM, 54H05DM 54S05DM, 54S05ADM 54LS05DM	6A
Flatpak (F)	A	74S05FC, 74S05AFC 74LS05FC	54S05FM, 54S05AFM 54LS05FM	3I
	B	7405FC, 74H05FC	5405FM, 54H05FM	

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74H (U.L.) HIGH/LOW	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	1.25/1.25	1.25/1.25	0.5/0.25
Outputs	OC**/10	OC**/12.5	OC**/12.5	OC**/5.0 (2.5)

DC AND AC CHARACTERISTICS: See Section 3*

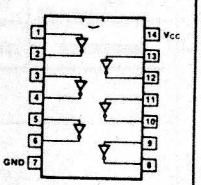
SYMBOL	PARAMETER	54/74 Min	54/74 Max	54/74H Min	54/74H Max	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS	
ICCH	Power Supply	12		26		19.8		2.4		mA	VIN = Gnd	VCC = Max
ICCL	Current	33		58		54		6.6			VIN = Open	
tPLH	Propagation Delay	55		18		2.0	7.5	22		ns	Fig. 3-2, 3-4	
tPHL		15		15		2.0	7.0	18				
tPLH	Propagation Delay (54S/74S05A only)					2.0	5.5			ns	Fig. 3-2, 3-4	
tPHL						1.5	5.0					

*DC limits apply over operating temperature range; AC limits apply at TA = +25°C and VCC = +5.0 V.
**OC — Collector

06

54/7406
HEX INVERTER BUFFER/DRIVER
(With Open-Collector High-Voltage Output)

CONNECTION DIAGRAM
PINOUT A



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7406PC		9A
Ceramic DIP (D)	A	7406DC	5406DM	6A
Flatpak (F)	A	7406FC	5406FM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW
Inputs	1.0/1.0
Outputs	OC**/10

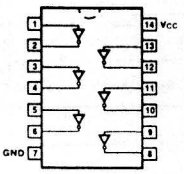
DC AND AC CHARATERISTICS: See Section 3*

SYMBOL	PARAMETER		54/74 Min	54/74 Max	UNITS	CONDITIONS	
VOL	Output LOW Voltage	XC		0.7	V	IOL = 40 mA	VCC = Min
		XM		0.7		IOL = 30 mA	VIN = VIH
		XC, XM		0.4		IOL = 16 mA	
IOH	Output HIGH Current			0.25	mA	VOH = 30 V, VIN = VIL	VCC = Min
ICCH	Power Supply Current			48	mA	VIN = Gnd	VCC = Max
ICCL				51		VIN = Open	
tPLH	Propagation Delay			15	ns	Fig. 3-2, 3-4	
tPHL				23			

*DC limits apply over operating temperature range; AC limits apply at TA = +25°C and VCC = +5.0 V.
**OC — Open Collector

54/7414
54LS/74LS14
HEX SCHMITT TRIGGER INVERTER

CONNECTION DIAGRAM
PINOUT A



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7414PC, 74LS14PC		9A
Ceramic DIP (D)	A	7414DC, 74LS14DC	5414DM, 54LS14DM	6A
Flatpak (F)	A	7414FC, 74LS14FC	5414FM, 54LS14FM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	0.5/0.25
Outputs	20/10	10/5.0 (2.5)

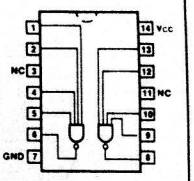
DC AND AC CHARACTERISTICS: See Section 3*

SYMBOL	PARAMETER	54/74 Min	54/74 Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS	
V_{T+}	Positive-going Threshold Voltage	1.5	2.0	1.5	2.0	V	V_{CC} = +5.0 V	
V_{T-}	Negative-going Threshold Voltage	0.6	1.1	0.6	1.1	V	V_{CC} = +5.0 V	
$V_{T+} - V_{T-}$	Hysteresis Voltage	0.4		0.4		V	V_{CC} = +5.0 V	
I_{T+}	Input Current at Positive-going Threshold	-0.43**		-0.14**		mA	V_{CC} = +5.0 V, V_{IN} = V_{T+}	
I_{T-}	Input Current at Negative-going Threshold	-0.56**		-0.18**		mA	V_{CC} = +5.0 V, V_{IN} = V_{T-}	
I_{IL}	Input LOW Current		-1.2		-0.4	mA	V_{CC} = Max, V_{IN} = 0.4 V	
I_{OS}	Output Short Circuit Current	-18	-55	-20	-100	mA	V_{CC} = Max, V_{OUT} = 0 V	
I_{CCH}	Power Supply Current		36		16	mA	V_{IN} = Gnd	V_{CC} = Max
I_{CCL}			60		21		V_{IN} = Open	
t_{PLH} / t_{PHL}	Propagation Delay		22 / 22		22 / 22	ns	Figs. 3-1, 3-15	

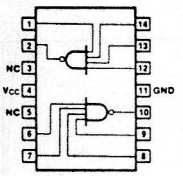
*DC limits apply over operating temperature range; AC limits apply at T_A = +25°C and V_{CC} = +5.0 V. **Typical Value

54/7420
54H/74H20
54S/74S20
54LS/74LS20
DUAL 4-INPUT NAND GATE

CONNECTION DIAGRAMS
PINOUT A



PINOUT B



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7420PC, 74H20PC, 74S20PC, 74LS20PC		9A
Ceramic DIP (D)	A	7420DC, 74H20DC, 74S20DC, 74LS20DC	5420DM, 54H20DM, 54S20DM, 54LS20DM	6A
Flatpak (F)	A	74S20FC, 74LS20FC	54S20FM, 54LS20FM	3I
	B	7420FC, 74H20FC	5420FM, 54H20FM	

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74H (U.L.) HIGH/LOW	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	1.25/1.25	1.25/1.25	0.5/0.25
Outputs	20/10	12.5/12.5	25/12.5	10/5.0 (2.5)

DC AND AC CHARACTERISTICS: See Section 3*

SYMBOL	PARAMETER	54/74 Min	54/74 Max	54/74H Min	54/74H Max	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS	
I_{CCH}	Power Supply	4.0		8.4		8.0		0.8		mA	V_{IN} = Gnd	V_{CC} = Max
I_{CCL}	Current	11		20		18		2.2			V_{IN} = Open	
t_{PLH} / t_{PHL}	Propagation Delay	22 / 15		10 / 10		2.0 / 2.0	4.5 / 5.0	15 / 15		ns	Figs. 3-1, 3-4	

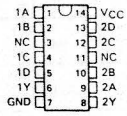
*DC limits apply over operating temperature range; AC limits apply at T_A = +25°C and V_{CC} = +5.0 V

SN54ALS20A, SN54AS20, SN74ALS20A, SN74AS20
DUAL 4-INPUT POSITIVE-NAND GATES

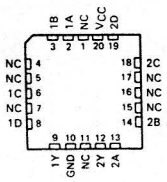
D2661, APRIL 1982 - REVISED MAY 1986

- **Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers, and Standard Plastic and Ceramic 300-mil DIPs**
- **Dependable Texas Instruments Quality and Reliability**

SN54ALS20A, SN54AS20 . . . J PACKAGE
SN74ALS20A, SN74AS20 . . . D OR N PACKAGE
(TOP VIEW)



SN54ALS20A, SN54AS20 . . . FK PACKAGE
(TOP VIEW)



NC—No internal connection

description

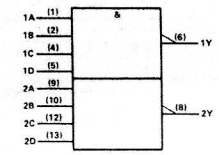
These devices contain two independent 4-input NAND gates. They perform the Boolean functions $Y = \overline{A \cdot B \cdot C \cdot D}$ or $Y = \overline{A} + \overline{B} + \overline{C} + \overline{D}$ in positive logic.

The SN54ALS20A and SN54AS20 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN74ALS20A and SN74AS20 are characterized for operation from 0°C to 70°C.

FUNCTION TABLE (each gate)

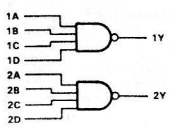
INPUTS A	B	C	D	OUTPUT Y
H	H	H	H	L
L	X	X	X	H
X	L	X	X	H
X	X	L	X	H
X	X	X	L	H

logic symbol†



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, and N packages.

logic diagram (positive logic)



PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1982, Texas Instruments Incorporated

TEXAS INSTRUMENTS
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

SN54ALS20A, SN74ALS20A
DUAL 4-INPUT POSITIVE-NAND GATES

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} . . . . . 7 V
Input voltage . . . . . 7 V
Operating free-air temperature range: SN54ALS20A . . . . . −55°C to 125°C
SN74ALS20A . . . . . 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C

recommended operating conditions

		SN54ALS20A MIN	NOM	MAX	SN74ALS20A MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High-level input voltage	2			2			V
V_{IL}	Low-level input voltage			0.7			0.8	V
I_{OH}	High-level output current			-0.4			-0.4	mA
I_{OL}	Low-level output current			4			8	mA
T_A	Operating free-air temperature	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS		SN54ALS20A MIN	TYP†	MAX	SN74ALS20A MIN	TYP†	MAX	UNIT
V_{IK}	V_{CC} = 4.5 V,	I_I = −18 mA			-1.5			-1.5	V
V_{OH}	V_{CC} = 4.5 V to 5.5 V,	I_{OH} = −0.4 mA	V_{CC}-2			V_{CC}-2			V
V_{OL}	V_{CC} = 4.5 V,	I_{OL} = 4 mA		0.25	0.4		0.25	0.4	V
	V_{CC} = 4.5 V,	I_{OL} = 8 mA					0.35	0.5	
I_I	V_{CC} = 5.5 V,	V_I = 7 V			0.1			0.1	mA
I_{IH}	V_{CC} = 5.5 V,	V_I = 2.7 V			20			20	μA
I_{IL}	V_{CC} = 5.5 V,	V_I = 0.4 V			-0.1			-0.1	mA
I_O‡	V_{CC} = 5.5 V,	V_O = 2.25 V	-30		-112	-30		-112	mA
I_{CCH}	V_{CC} = 5.5 V,	V_I = 0 V		0.22	0.4		0.22	0.4	mA
I_{CCL}	V_{CC} = 5.5 V,	V_I = 4.5 V		0.81	1.5		0.81	1.5	mA

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The output conditions have been chosen to produce a current that closely approximates one half of the true short-circuit output current, I_{OS}.

switching characteristics (see Note 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC} = 5 V, C_L = 50 pF, R_L = 500 Ω, T_A = 25°C 'ALS20A TYP	V_{CC} = 4.5 V to 5.5 V, C_L = 50 pF, R_L = 500 Ω, T_A = MIN to MAX SN54ALS20A MIN	SN54ALS20A MAX	SN74ALS20A MIN	SN74ALS20A MAX	UNIT
t_{PLH}	Any	Y	7	1	18	3	11	ns
t_{PHL}	Any	Y	6	1	15	3	10	

NOTE 1: Load circuit and voltage waveforms are shown in Section 1.

TEXAS INSTRUMENTS
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} . . . . . 7 V
Input voltage . . . . . 7 V
Operating free-air temperature range: SN54AS20 . . . . . −55°C to 125°C
SN74AS20 . . . . . 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C

recommended operating conditions

		SN54AS20			SN74AS20			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage	4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High-level input voltage	2			2			V
V_{IL}	Low-level input voltage			0.8			0.8	V
I_{OH}	High-level output current			−2			−2	mA
I_{OL}	Low-level output current			20			20	mA
T_A	Operating free-air temperature	−55		125	0		70	°C

electrical characteristics over recommended operating-free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS		SN54AS20			SN74AS20			UNIT
			MIN	TYP†	MAX	MIN	TYP†	MAX	
V_{IK}	V_{CC} = 4.5 V,	I_I = −18 mA			−1.2			−1.2	V
V_{OH}	V_{CC} = 4.5 V to 5.5 V,	I_{OH} = −2 mA	V_{CC} − 2			V_{CC} − 2			V
V_{OL}	V_{CC} = 4.5 V,	I_{OL} = 20 mA		0.35	0.5		0.35	0.5	V
I_I	V_{CC} = 5.5 V,	V_I = 7 V			0.1			0.1	mA
I_{IH}	V_{CC} = 5.5 V,	V_I = 2.7 V			20			20	µA
I_{IL}	V_{CC} = 5.5 V,	V_I = 0.4 V			−0.5			−0.5	mA
I_O‡	V_{CC} = 5.5 V,	V_O = 2.25 V	−30		−112	−30		−112	mA
I_{CCH}	V_{CC} = 5.5 V,	V_I = 0 V		1	1.6		1	1.6	mA
I_{CCL}	V_{CC} = 5.5 V,	V_I = 4.5 V		5.4	8.7		5.4	8.7	mA

† All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡ The output conditions have been chosen to produce a current that closely approximates one half of the true short-circuit output current, I_{OS}.

switching characteristics (see Note 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC} = 4.5 V to 5.5 V, C_L = 50 pF, R_L = 500 Ω, T_A = MIN to MAX				UNIT
			SN54AS20		SN74AS20		
			MIN	MAX	MIN	MAX	
t_{PLH}	Any	Y	1	5.5	1	5	ns
t_{PHL}	Any	Y	1	5	1	4.5	ns

NOTE 1: Load circuit and voltage waveforms are shown in Section 1.

recommended operating conditions

		SN54LS20			SN74LS20			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage	4.5	5	5.5	4.75	5	5.25	V
V_{IH}	High-level input voltage	2			2			V
V_{IL}	Low-level input voltage			0.7			0.8	V
I_{OH}	High-level output current			−0.4			−0.4	mA
I_{OL}	Low-level output current			4			8	mA
T_A	Operating free-air temperature	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS †		SN54LS20			SN74LS20			UNIT
			MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IK}	V_{CC} = MIN,	I_I = −18 mA			−1.5			−1.5	V
V_{OH}	V_{CC} = MIN,	V_{IL} = MAX, I_{OH} = −0.4 mA	2.5	3.4		2.7	3.4		V
V_{OL}	V_{CC} = MIN,	V_{IH} = 2 V, I_{OL} = 4 mA		0.25	0.4			0.4	V
	V_{CC} = MIN,	V_{IH} = 2 V, I_{OL} = 8 mA					0.25	0.5	
I_I	V_{CC} = MAX,	V_I = 7 V			0.1			0.1	mA
I_{IH}	V_{CC} = MAX,	V_I = 2.7 V			20			20	µA
I_{IL}	V_{CC} = MAX,	V_I = 0.4 V			−0.4			−0.4	mA
I_{OS}§	V_{CC} = MAX		−20		−100	−20		−100	mA
I_{CCH}	V_{CC} = MAX,	V_I = 0 V		0.4	0.8		0.4	0.8	mA
I_{CCL}	V_{CC} = MAX,	V_I = 4.5 V		1.2	2.2		1.2	2.2	mA

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡ All typical values are at V_{CC} = 5 V, T_A = 25°C.
§ Not more than one output should be shorted at a time, and the duration of the short-circuit should not exceed one second.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see note 2)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{PLH}	Any	Y	R_L = 2 kΩ, C_L = 15 pF		9	15	ns
t_{PHL}					10	15	ns

NOTE 2: See General Information Section for load circuits and voltage waveforms.

TEXAS INSTRUMENTS
POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

HIGH-SPEED CMOS LOGIC

TABLE I
SPECIFICATIONS FOR HC SSI CIRCUITS

D2804, DECEMBER 1982 – REVISED MARCH 1984

absolute maximum ratings over operating free-air temperature range†

Supply voltage range, V_{CC} . . . . . −0.5 V to 7 V
Input diode current, I_{IK} (V_I < 0 or V_I > V_{CC}) . . . . . ±20 mA
Output diode current, I_{OK} (V_O < 0 or V_O > V_{CC}) . . . . . ±20 mA
Continuous output current, I_O (V_O = 0 to V_{CC}) . . . . . ±25 mA
Continuous current through V_{CC} or GND pins . . . . . ±50 mA
Lead temperature 1,6 mm (1/16 inch) from case for 60 seconds: FH, FK, or J package . . . . 300°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds: FN or N package . . . . . 260°C
Storage temperature range . . . . . −65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

recommended operating conditions

			SN54HC'			SN74HC'			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage		2	5	6	2	5	6	V
V_{IH}	High-level input voltage	V_{CC} = 2 V	1.5			1.5			V
		V_{CC} = 4.5 V	3.15			3.15			
		V_{CC} = 6 V	4.2			4.2			
V_{IL}	Low-level input voltage	V_{CC} = 2 V	0		0.3	0		0.3	V
		V_{CC} = 4.5 V	0		0.9	0		0.9	
		V_{CC} = 6 V	0		1.2	0		1.2	
V_I	Input voltage		0		V_{CC}	0		V_{CC}	V
V_O	Output voltage		0		V_{CC}	0		V_{CC}	V
t_t	Input transition (rise and fall) times (except Schmitt-trigger inputs)	V_{CC} = 2 V	0		1000	0		1000	ns
		V_{CC} = 4.5 V	0		500	0		500	
		V_{CC} = 6 V	0		400	0		400	
T_A	Operating free-air temperature		−55		125	−40		85	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS	V_{CC}	T_A = 25°C			SN54HC'		SN74HC'		UNIT
			MIN	TYP	MAX	MIN	MAX	MIN	MAX	
V_{OH} (Totem-pole outputs)	V_I = V_{IH} or V_{IL}, I_{OH} = −20 µA	2 V	1.9	1.998		1.9		1.9		V
		4.5 V	4.4	4.499		4.4		4.4		
		6 V	5.9	5.999		5.9		5.9		
	V_I = V_{IH} or V_{IL}, I_{OH} = −4 mA	4.5 V	3.98	4.30		3.7		3.84		
	V_I = V_{IH} or V_{IL}, I_{OH} = −5.2 mA	6 V	5.48	5.80		5.2		5.34		
I_{OH} (Open-drain outputs)	V_I = V_{IH} or V_{IL}, V_O = V_{CC}	6 V		0.01	0.5		10		5	µA
V_{OL}	V_I = V_{IH} or V_{IL}, I_{OL} = 20 µA	2 V		0.002	0.1		0.1		0.1	V
		4.5 V		0.001	0.1		0.1		0.1	
		6 V		0.001	0.1		0.1		0.1	
	V_I = V_{IH} or V_{IL}, I_{OL} = 4 mA	4.5 V		0.17	0.26		0.4		0.33	
	V_I = V_{IH} or V_{IL}, I_{OL} = 5.2 mA	6 V		0.15	0.26		0.4		0.33	
V_{T+}†		2 V	0.8	1.2	1.5					V
		4.5 V	2	2.5	3.15					
		6 V	2.5	3.3	4.2					
V_{T-}†		2 V	0.3	0.6	0.8					V
		4.5 V	0.9	1.6	2					
		6 V	1.2	2	2.5					
$V_{T+} - V_{T-}$†		2 V	0.2	0.6	1					V
		4.5 V	0.4	0.9	1.4					
		6 V	0.5	1.3	1.7					
I_I	V_I = 0 to V_{CC}	6 V		±0.1	±100		±1000		±1000	nA
I_{CC}	V_I = V_{CC} or 0, I_O = 0	6 V			2		40		20	µA
C_I		2 to 6 V		3	10		10		10	pF

† This parameter applies only for Schmitt-trigger inputs.

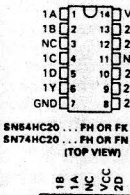
HIGH-SPEED CMOS LOGIC

TYPES SN54HC20, SN74HC20
DUAL 4-INPUT POSITIVE-NAND GATES

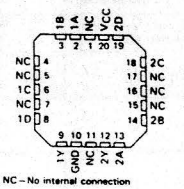
D2684, DECEMBER 1982 – REVISED MARCH 1984

- **Package Options Include Both Plastic and Ceramic Chip Carriers in Addition to Plastic and Ceramic DIPs**
- **Dependable Texas Instruments Quality and Reliability**

SN54HC20 . . . J PACKAGE
SN74HC20 . . . J OR N PACKAGE
(TOP VIEW)



SN54HC20 . . . FH OR FK PACKAGE
SN74HC20 . . . FH OR FN PACKAGE
(TOP VIEW)



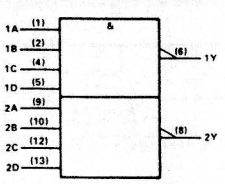
NC – No internal connection

description

These devices contain two independent 4-input NAND gates. They perform the Boolean functions $Y = \overline{A \cdot B \cdot C \cdot D}$ or $Y = \overline{A} + \overline{B} + \overline{C} + \overline{D}$ in positive logic.

The SN54HC20 is characterized for operation over the full military temperature range of −55°C to 125°C. The SN74HC20 is characterized for operation from −40°C to 85°C.

logic symbol



Pin numbers shown are for J and N packages.

FUNCTION TABLE (each gate)

INPUTS				OUTPUT
A	B	C	D	Y
H	H	H	H	L
L	X	X	X	H
X	L	X	X	H
X	X	L	X	H
X	X	X	L	H

maximum ratings, recommended operating conditions, and electrical characteristics

See Table I, page 2-4.

switching characteristics over recommended operating free-air temperature range (unless otherwise noted), C_L = 50 pF (see Note 1)

PARAMETER	FROM (INPUT)	TO (OUTPUT)	V_{CC}	T_A = 25°C			SN54HC20		SN74HC20		UNIT
				MIN	TYP	MAX	MIN	MAX	MIN	MAX	
t_{pd}	A, B, C, or D	Y	2 V		45	110		165		140	ns
			4.5 V		14	22		33		28	
			6 V		11	19		28		24	
t_t		Y	2 V		27	75		110		95	ns
			4.5 V		9	15		22		19	
			6 V		7	13		19		16	

C_{pd}	Power dissipation capacitance per gate	No load, T_A = 25°C	25 pF typ

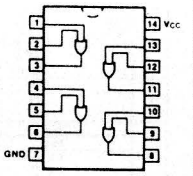
NOTE 1: For load circuit and voltage waveforms, see page 1-14.

Copyright ©1982 by Texas Instruments Incorporated

TEXAS INSTRUMENTS
POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

54/7432
54S/74S32
54LS/74LS32
QUAD 2-INPUT OR GATE

CONNECTION DIAGRAM
PINOUT A



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7432PC, 74S32PC, 74LS32PC		9A
Ceramic DIP (D)	A	7432DC, 74S32DC, 74LS32DC	5432DM, 54S32DM, 54LS32DM	6A
Flatpak (F)	A	7432FC, 74S32FC, 74LS32FC	5432FM, 54S32FM, 54LS32FM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	1.25/1.25	0.5/0.25
Outputs	20/10	25/12.5	10/5.0 (2.5)

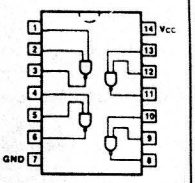
DC AND AC CHARACTERISTICS: See Section 3 for U.L. definitions

SYMBOL	PARAMETER	54/74 Min	54/74 Max	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS	
ICCH	Power Supply Current		22		32		6.2	mA	VIN = Open	VCC = Max
ICCL			38		68		9.8		VIN = Gnd	
tPLH	Propagation Delay		15	2.0	7.0		15	ns	Figs. 3-1, 3-5	
tPHL			22	2.0	7.0		15			

*DC limits apply over operating temperature range; AC limits apply at TA = +25°C and VCC = +5.0 V.

54/7437
54LS/74LS37
QUAD 2-INPUT NAND BUFFER

CONNECTION DIAGRAM
PINOUT A



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7437PC, 74LS37PC		9A
Ceramic DIP (D)	A	7437DC, 74LS37DC	5437DM, 54LS37DM	6A
Flatpak (F)	A	7437FC, 74LS37FC	5437FM, 54LS37FM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	0.5/0.25
Outputs	30/30	30/15 (7.5)

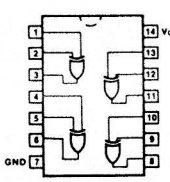
DC AND AC CHARACTERISTICS: See Section 3*

SYMBOL	PARAMETER		54/74 Min	54/74 Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS	
VOH	Output HIGH Voltage	XM	2.4		2.5		V	VCC = Max, IOH = -1.2 mA	
		XC	2.4		2.7			VIN = VIL	
VOL	Output LOW Voltage	XM, XC		0.4			V	IOL = 48 mA	VCC = Min, VIN = 2.0 V
		XM				0.4		IOL = 12 mA	
		XC				0.5		IOL = 24 mA	
IOS	Output Short Circuit Current	XM	-20	-70	-30	-130	mA	VCC = Min, VOUT = 0 V	
		XC	-18	-70	-30	-130			
ICCH	Power Supply Current			15.5		2.0	mA	VIN = Gnd	VCC = Max
ICCL				54		12		VIN = Open	
tPLH	Propagation Delay			22		20	ns	Figs. 3-1, 3-4	
tPHL				15		20			

*DC limits apply over operating temperature range; AC limits apply at TA = +25°C and VCC = +5.0 V.

54/7486
54S/74S86
54LS/74LS86
QUAD 2-INPUT EXCLUSIVE-OR GATE

CONNECTION DIAGRAM
PINOUT A



ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: VCC = +5.0 V ±5%, TA = 0°C to +70°C	MILITARY GRADE: VCC = +5.0 V ±10%, TA = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	7486PC, 74S86PC, 74LS86PC		9A
Ceramic DIP (D)	A	7486DC, 74S86DC, 74LS86DC	5486DM, 54S86DM, 54LS86DM	6A
Flatpak (F)	A	7486FC, 74S86FC, 74LS86FC	5486FM, 54S86FM, 54LS86FM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PINS	54/74 (U.L.) HIGH/LOW	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
Inputs	1.0/1.0	1.25/1.25	1.0/0.375
Outputs	20/10	25/12.5	10/5.0 (2.5)

DC AND AC CHARACTERISTICS: See Section 3*

SYMBOL	PARAMETER		54/74 Min	54/74 Max	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS
ICC	Power Supply Current	XM		43		75		10	mA	VCC = Max, VIN = Gnd
		XC		50		75		10		
tPLH	Propagation Delay			23	3.5	10.5		12	ns	Other Input LOW
tPHL				17	3.0	10		17		Figs. 3-1, 3-5
tPLH	Propagation Delay			30	3.5	10.5		13	ns	Other Input HIGH
tPHL				22	3.0	10		12		Figs. 3-1, 3-4

*DC limits apply over operating temperature range; AC limits apply at TA = +25°C and VCC = +5.0 V.

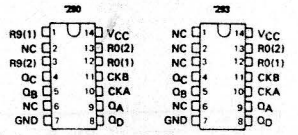
SN54290, SN54293, SN54LS290, SN54LS293, SN74290, SN74293, SN74LS290, SN74LS293
DECADE AND 4-BIT BINARY COUNTERS

MARCH 1974 – REVISED MARCH 1988

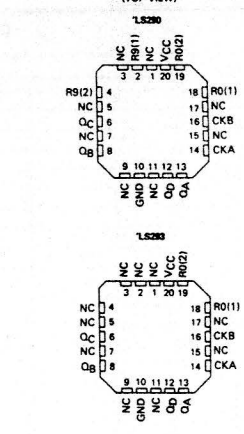
'290, 'LS290 . . . DECADE COUNTERS
'293, 'LS293 . . . 4-BIT BINARY COUNTERS

- GND and VCC on Corner Pins (Pins 7 and 14 Respectively)

SN54290, SN54LS290, SR54293, SN54LS293 . . . J OR W PACKAGE
SN74290, SN74293 . . . N PACKAGE
SN74LS290, SN74LS293 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS290, SN54LS293 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

description

The SN54290/SN74290, SN54LS290/SN74LS290, SN54293/SN74293, and SN54LS293/SN74LS293 counters are electrically and functionally identical to the SN5490A/SN7490A, SN54LS90/SN74LS90, SN5493A/SN7493A, and SN54LS93/SN74LS93, respectively. Only the arrangement of the terminals has been changed for the '290, 'LS290, '293, and 'LS293.

Each of these monolithic counters contains four master-slave flip-flops and additional gating to provide a divide-by-two counter and a three-stage binary counter for which the count cycle length is divide-by-five for the '290 and 'LS290 and divide-by-eight for the '293 and 'LS293.

All of these counters have a gated zero reset and the '290 and 'LS290 also have gated set-to-nine inputs for use in BCD nine's complement applications.

To use the maximum count length (decade or four-bit binary) of these counters, the B input is connected to the Q_A output. The input count pulses are applied to input A and the outputs are as described in the appropriate function table. A symmetrical divide-by-ten count can be obtained from the '290 and 'LS290 counters by connecting the Q_D output to the A input and applying the input count to the B input which gives a divide-by-ten square wave at output Q_A.

PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

TEXAS INSTRUMENTS
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

'290, 'LS290 BCD COUNT SEQUENCE (See Note A)

COUNT	OUTPUT Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	H
9	H	L	L	L

'290, 'LS290 BI-QUINARY (5-2) (See Note B)

COUNT	OUTPUT Q_A	Q_D	Q_C	Q_B
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	H	L	L	L
6	H	L	L	H
7	H	L	H	L
8	H	L	H	H
9	H	H	L	L

'290, 'LS290 RESET/COUNT FUNCTION TABLE

RESET INPUTS $R_{0(1)}$	$R_{0(2)}$	$R_{9(1)}$	$R_{9(2)}$	OUTPUT Q_D	Q_C	Q_B	Q_A
H	H	L	X	L	L	L	L
H	H	X	L	L	L	L	L
X	X	H	H	H	L	L	H
X	L	X	L	COUNT			
L	X	L	X	COUNT			
L	X	X	L	COUNT			
X	L	L	X	COUNT			

'293, 'LS293 COUNT SEQUENCE (See Note C)

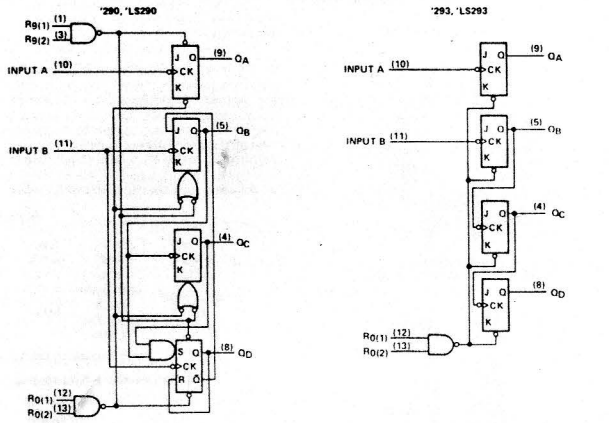
COUNT	OUTPUT Q_D	Q_C	Q_B	Q_A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

'293, 'LS293 RESET/COUNT FUNCTION TABLE

RESET INPUTS $R_{0(1)}$	$R_{0(2)}$	OUTPUT Q_D	Q_C	Q_B	Q_A
H	H	L	L	L	L
L	X	COUNT			
X	L	COUNT			

NOTES: A. Output Q_A is connected to input B for BCD count.
B. Output Q_D is connected to input A for bi-quinary count.
C. Output Q_A is connected to input B.
D. H = high level, L = low level, X = irrelevant

logic diagrams (positive logic)



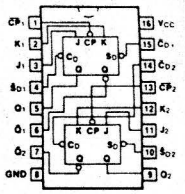
Pin numbers shown are for D, J, N, and W packages.
The J and K inputs shown without connection are for reference only and are functionally at a high level.

TEXAS INSTRUMENTS
POST OFFICE BOX 655012 • DALLAS, TEXAS 75265

54S/74S112
54LS/74LS112
DUAL JK NEGATIVE EDGE-TRIGGERED FLIP-FLOP

DESCRIPTION — The '112 features individual J, K, Clock and asynchronous Set and Clear inputs to each flip-flop. When the clock goes HIGH, the inputs are enabled and data will be accepted. The logic level of the J and K inputs may change when the clock is HIGH and the bistable will perform according to the Truth Table as long as minimum setup and hold times are observed. Input data is transferred to the outputs on the falling edge of the clock pulse.

CONNECTION DIAGRAM
PINOUT A



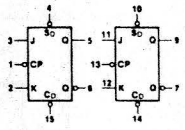
TRUTH TABLE

INPUTS @ t_n J	K	OUTPUT @ t_{n+1} Q
L	L	Q_n
L	H	L
H	L	H
H	H	$\overline{Q}_n$

Asynchronous Inputs:
LOW input to $\overline{S}_D$ sets Q to HIGH level
LOW input to $\overline{C}_D$ sets Q to LOW level
Clear and Set are independent of clock
Simultaneous LOW on $\overline{C}_D$ and $\overline{S}_D$ makes both Q and $\overline{Q}$ HIGH

t_n = Bit time before clock pulse
t_{n+1} = Bit time after clock pulse
H = HIGH Voltage Level
L = LOW Voltage Level

LOGIC SYMBOL



V_{CC} = Pin 16
GND = Pin 8

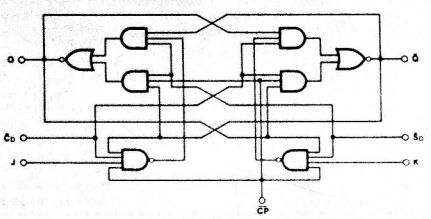
ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE V_{CC} = +5.0 V ±5%, T_A = 0°C to +70°C	MILITARY GRADE V_{CC} = +5.0 V ±10%, T_A = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	74S112PC, 74LS112PC		9B
Ceramic DIP (D)	A	74S112DC, 74LS112DC	54S112DM, 54LS112DM	6B
Flatpak (F)	A	74S112FC, 74LS112FC	54S112FM, 54LS112FM	4L

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PIN NAMES	DESCRIPTION	54/74S (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
J_1, J_2, K_1, K_2	Data Inputs	1.25/1.0	0.5/0.25
$\overline{CP}_1$, $\overline{CP}_2$	Clock Pulse Inputs (Active Falling Edge)	2.5/2.5	2.0/0.5
$\overline{C}_{D1}$, $\overline{C}_{D2}$	Direct Clear Inputs (Active LOW)	2.5/4.375	1.5/0.5
$\overline{S}_{D1}$, $\overline{S}_{D2}$	Direct Set Inputs (Active LOW)	2.5/4.375	1.5/0.5
Q_1, Q_2, $\overline{Q}_1$, $\overline{Q}_2$	Outputs	25/12.5	10/5.0 (2.5)

LOGIC DIAGRAM (one half shown)



DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

SYMBOL	PARAMETER	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS
I_{CC}	Power Supply Current		50		8.0	mA	V_{CC} = Max, V_{CP} = 0 V

AC CHARACTERISTICS: V_{CC} = +5.0 V, T_A = +25°C (See Section 3 for waveforms and load configurations)

SYMBOL	PARAMETER	54/74S C_L = 15 pF, R_L = 280 Ω Min	54/74S Max	54/74LS C_L = 15 pF Min	54/74LS Max	UNITS	CONDITIONS
f_{max}	Maximum Clock Frequency	80		30		MHz	Figs. 3-1, 3-9
t_{PLH} t_{PHL}	Propagation Delay $\overline{CP}_n$ to Q_n or $\overline{Q}_n$		7.0 7.0		16 24	ns	Figs. 3-1, 3-9
t_{PLH} t_{PHL}	Propagation Delay $\overline{C}_{Dn}$ or $\overline{S}_{Dn}$ to Q_n or $\overline{Q}_n$		7.0 7.0		16 24	ns	Figs. 3-1, 3-10

AC OPERATING REQUIREMENTS: V_{CC} = +5.0 V, T_A = +25°C

SYMBOL	PARAMETER	54/74S Min	54/74S Max	54/74LS Min	54/74LS Max	UNITS	CONDITIONS
t_s (H) t_s (L)	Setup Time J_n or K_n to $\overline{CP}_n$		7.0 7.0		20 15	ns	Fig. 3-7
t_h (H) t_h (L)	Hold Time J_n or K_n to $\overline{CP}_n$		0 0		0 0	ns	Fig. 3-7
t_w (H) t_w (L)	$\overline{CP}_n$ Pulse Width		6.0 6.5		20 15	ns	Fig. 3-9
t_w (L)	$\overline{C}_{Dn}$ or $\overline{S}_{Dn}$ Pulse Width LOW		8.0		15	ns	Fig. 3-10

54/74121
MONOSTABLE MULTIVIBRATOR

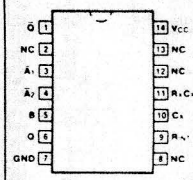
DESCRIPTION — The '121 features positive and negative dc level triggering inputs and complementary outputs. Input pin 5 directly activates a Schmitt circuit which provides temperature compensated level detection, increases immunity to positive-going noise and assures jitter-free response to slowly rising triggers.

When triggering occurs, internal feedback latches the circuit, prevents re-triggering while the output pulse is in progress and increases immunity to negative-going noise. Noise immunity is typically 1.2 V at the inputs and 1.5 V on V_{CC}.

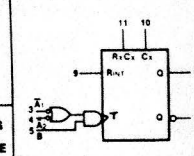
Output pulse width stability is primarily a function of the external R_x and C_x chosen for the application. A 2 kΩ internal resistor is provided for optional use where output pulse width stability requirements are less stringent. Maximum duty cycle capability ranges from 67% with a 2 kΩ resistor to 90% with a 40 kΩ resistor. Duty cycles beyond this range tend to reduce the output pulse width. Otherwise, output pulse width follows the relationship:

$$t_w = 0.69\ R_X C_X$$

CONNECTION DIAGRAM
PINOUT A



LOGIC SYMBOL



V_{CC} = Pin 14
GND = Pin 7
NC = Pins 2,8,12,13

ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE V_{CC} = +5.0 V ±5%, T_A = 0°C to +70°C	MILITARY GRADE V_{CC} = +5.0 V ±10%, T_A = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	74121PC		9A
Ceramic DIP (D)	A	74121DC	54121DM	6A
Flatpak (F)	A	74121FC	54121FM	3I

INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PIN NAMES	DESCRIPTION	54/74 (U.L.) HIGH/LOW
$\overline{A}_1$, $\overline{A}_2$	Trigger Inputs (Active Falling Edge)	1.0/1.0
B	Schmitt Trigger Input (Active Rising Edge)	2.0/2.0
Q, $\overline{Q}$	Outputs	20/10

54/74193
54LS/74LS193
UP/DOWN BINARY COUNTER
(With Separate Up/down Clocks)

DESCRIPTION — The '193 is an up/down modulo-16 binary counter. Separate Count Up and Count Down Clocks are used and in either counting mode the circuits operate synchronously. The outputs change state synchronous with the LOW-to-HIGH transitions on the clock inputs. Separate Terminal Count Up and Terminal Count Down outputs are provided which are used as the clocks for subsequent stages without extra logic, thus simplifying multistage counter designs. Individual preset inputs allow the circuits to be used as programmable counters. Both the Parallel Load ($\overline{PL}$) and the Master Reset (MR) inputs asynchronously override the clocks. For functional description and detail specifications please refer to the '192 data sheet.

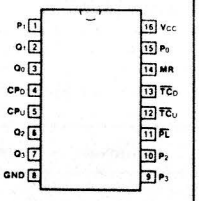
ORDERING CODE: See Section 9

PKGS	PIN OUT	COMMERCIAL GRADE: V_{CC} = +5.0 V ±5%, T_A = 0°C to +70°C	MILITARY GRADE: V_{CC} = +5.0 V ±10%, T_A = -55°C to +125°C	PKG TYPE
Plastic DIP (P)	A	74193PC, 74LS193PC		9B
Ceramic DIP (D)	A	74193DC, 74LS193DC	54193DM, 54LS193DM	6B
Flatpak (F)	A	74193FC, 74LS193FC	54193FM, 54LS193FM	4L

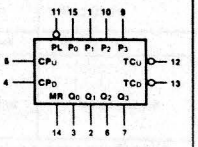
INPUT LOADING/FAN-OUT: See Section 3 for U.L. definitions

PIN NAMES	DESCRIPTION	54/74 (U.L.) HIGH/LOW	54/74LS (U.L.) HIGH/LOW
CP_U	Count Up Clock Input (Active Rising Edge)	1.0/1.0	0.5/0.25
CP_D	Count Down Clock Input (Active Rising Edge)	1.0/1.0	0.5/0.25
MR	Asynchronous Master Reset Input (Active HIGH)	1.0/1.0	0.5/0.25
$\overline{PL}$	Asynchronous Parallel Load Input (Active LOW)	1.0/1.0	0.5/0.25
$P_0 - P_3$	Parallel Data Inputs	1.0/1.0	0.5/0.25
$Q_0 - Q_3$	Flip-flop Outputs	20/10	10/5.0 (2.5)
$\overline{TC}_D$	Terminal Count Down (Borrow) Output (Active LOW)	20/10	10/5.0 (2.5)
$\overline{TC}_U$	Terminal Count Up (Carry) Output (Active LOW)	20/10	10/5.0 (2.5)

CONNECTION DIAGRAM
PINOUT A



LOGIC SYMBOL



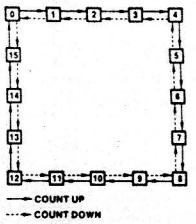
VCC = Pin 16
GND = Pin 8

MODE SELECT TABLE

MR	$\overline{PL}$	CP_U	CP_D	MODE
H	X	X	X	Reset (Asyn.)
L	L	X	X	Preset (Asyn.)
L	H	H	H	No Change
L	H	⌐	H	Count Up
L	H	H	⌐	Count Down

H = HIGH Voltage Level
L = LOW Voltage Level
X = Immaterial
Z = High Impedance

STATE DIAGRAM

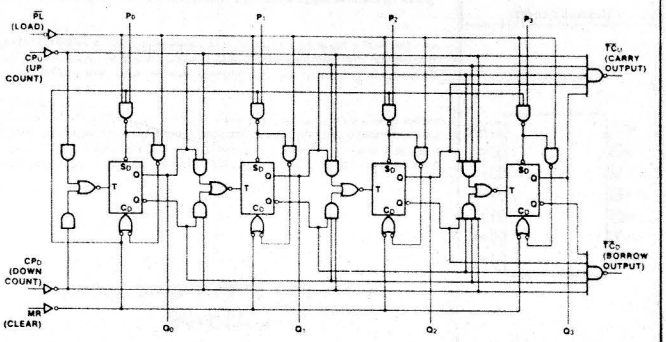


LOGIC EQUATIONS FOR TERMINAL COUNT

$\overline{TC}_U = Q_0 \cdot Q_1 \cdot Q_2 \cdot Q_3 \cdot \overline{CP}_U$

$\overline{TC}_D = \overline{Q}_0 \cdot \overline{Q}_1 \cdot \overline{Q}_2 \cdot \overline{Q}_3 \cdot \overline{CP}_D$

LOGIC DIAGRAM



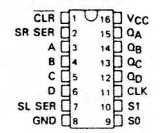
SN54194, SN54LS194A, SN54S194, SN74194, SN74LS194A, SN74S194
4-BIT BIDIRECTIONAL UNIVERSAL SHIFT REGISTERS

MARCH 1974 – REVISED MARCH 1988

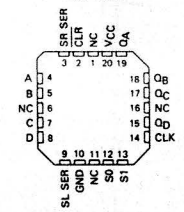
- Parallel Inputs and Outputs
- Four Operating Modes:
  - Synchronous Parallel Load
  - Right Shift
  - Left Shift
  - Do Nothing
- Positive Edge-Triggered Clocking
- Direct Overriding Clear

TYPE	TYPICAL MAXIMUM CLOCK FREQUENCY	TYPICAL POWER DISSIPATION
'194	36 MHz	195 mW
'LS194A	36 MHz	75 mW
'S194	105 MHz	425 mW

SN54194, SN54LS194A, SN54S194 . . . J OR W PACKAGE
SN74194 . . . N PACKAGE
SN74LS194A, SN74S194 . . . D OR N PACKAGE
(TOP VIEW)



SN54LS194A, SN54S194 . . . FK PACKAGE
(TOP VIEW)



NC – No internal connection

description

These bidirectional shift registers are designed to incorporate virtually all of the features a system designer may want in a shift register. The circuit contains 46 equivalent gates and features parallel inputs, parallel outputs, right-shift and left-shift serial inputs, operating-mode-control inputs, and a direct overriding clear line. The register has four distinct modes of operation, namely:

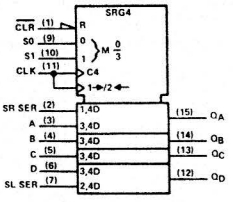
- Inhibit clock (do nothing)
- Shift right (in the direction Q_A toward Q_D)
- Shift left (in the direction Q_D toward Q_A)
- Parallel (broadside) load

Synchronous parallel loading is accomplished by applying the four bits of data and taking both mode control inputs, S0 and S1, high. The data are loaded into the associated flip-flops and appear at the outputs after the positive transition of the clock input. During loading, serial data flow is inhibited.

Shift right is accomplished synchronously with the rising edge of the clock pulse when S0 is high and S1 is low. Serial data for this mode is entered at the shift-right data input. When S0 is low and S1 is high, data shifts left synchronously and new data is entered at the shift-left serial input.

Clocking of the shift register is inhibited when both mode control inputs are low. The mode controls of the SN54194/SN74194 should be changed only while the clock input is high.

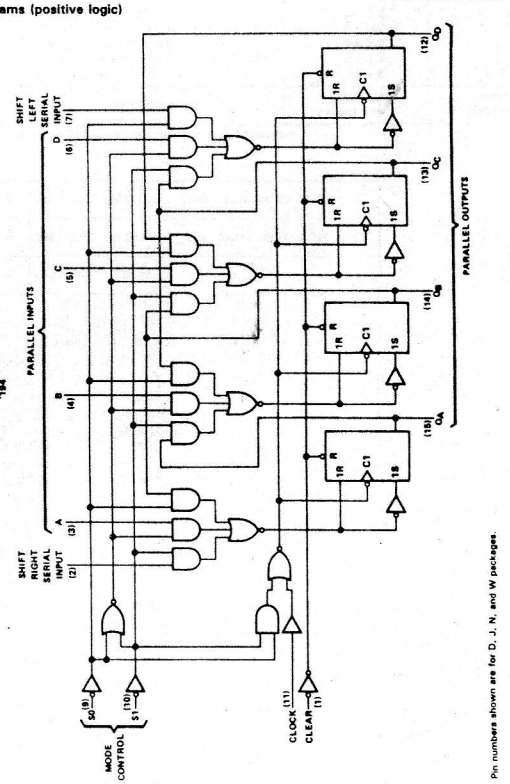
logic symbol†



†This symbol is in accordance with ANSI/IEEE Std. 91-1984 and IEC Publication 617-12.
Pin numbers shown are for D, J, N, and W packages.

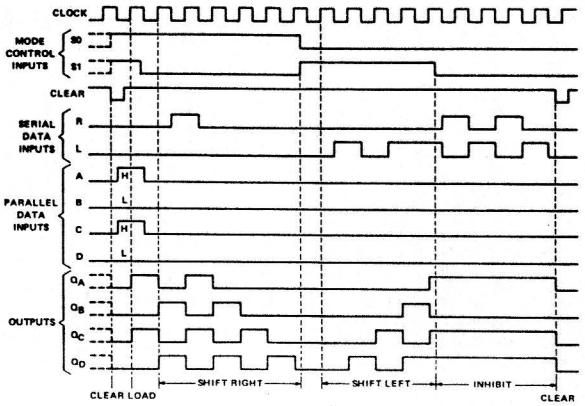
PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

logic diagrams (positive logic)



Pin numbers shown are for D, J, N, and W packages.

typical clear, load, right-shift, left-shift, inhibit, and clear sequences



FUNCTION TABLE

INPUTS											OUTPUTS			
CLEAR	MODE S1	MODE S0	CLOCK	SERIAL LEFT	SERIAL RIGHT	PARALLEL A	PARALLEL B	PARALLEL C	PARALLEL D		Q_A	Q_B	Q_C	Q_D
L	X	X	X	X	X	X	X	X	X		L	L	L	L
H	X	X	L	X	X	X	X	X	X		Q_{A0}	Q_{B0}	Q_{C0}	Q_{D0}
H	H	H	↑	X	X	a	b	c	d		a	b	c	d
H	L	H	↑	X	H	X	X	X	X		H	Q_{An}	Q_{Bn}	Q_{Cn}
H	L	H	↑	X	L	X	X	X	X		L	Q_{An}	Q_{Bn}	Q_{Cn}
H	H	L	↑	H	X	X	X	X	X		Q_{Bn}	Q_{Cn}	Q_{Dn}	H
H	H	L	↑	L	X	X	X	X	X		Q_{Bn}	Q_{Cn}	Q_{Dn}	L
H	L	L	X	X	X	X	X	X	X		Q_{An}	Q_{Bn}	Q_{Cn}	Q_{D0}

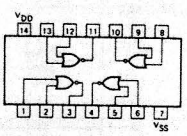
TEXAS INSTRUMENTS

4001B 4002B

QUAD 2-INPUT NOR GATE • DUAL 4-INPUT NOR GATE

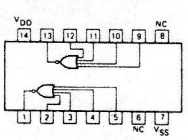
DESCRIPTION – These CMOS logic elements provide the positive input NOR function. The outputs are fully buffered for highest noise immunity and pattern insensitivity of output impedance.

4001B
LOGIC AND CONNECTION DIAGRAM
DIP (TOP VIEW)



NOTE:
The Flatpak versions have the same pinouts (Connection Diagram) as the Dual In-line Package.

4002B
LOGIC AND CONNECTION DIAGRAM
DIP (TOP VIEW)



DC CHARACTERISTICS: V_{DD} as shown, V_{SS} = 0 V

SYMBOL	PARAMETER		LIMITS V_{DD} = 5 V MIN	TYP	MAX	V_{DD} = 10 V MIN	TYP	MAX	V_{DD} = 15 V MIN	TYP	MAX	UNITS	TEMP	TEST CONDITIONS See Note 1
I_{DD}	Quiescent Power Supply Current	XC			1			2			4	μA	MIN, 25°C	All inputs at 0 V or V_{DD}
					7.5			15			30		MAX	
		XM			0.25			0.5			1	μA	MIN, 25°C	
					7.5			15			30		MAX	

AC CHARACTERISTICS: V_{DD} as shown, V_{SS} = 0 V, T_A = 25°C, 4001B only (See Note 2)

SYMBOL	PARAMETER	LIMITS V_{DD} = 5 V MIN	TYP	MAX	V_{DD} = 10 V MIN	TYP	MAX	V_{DD} = 15V MIN	TYP	MAX	UNITS	TEST CONDITIONS See Note 2
t_{PLH}	Propagation Delay		60	110		25	60		20	48	ns	C_L = 50 pF,
t_{PHL}			60	110		25	60		20	48	ns	R_L = 200 kΩ
t_{TLH}	Output Transition Time		60	135		30	70		20	45	ns	Input Transition
t_{THL}			60	135		30	70		20	45	ns	Times < 20 ns

AC CHARACTERISTICS: V_{DD} as shown, V_{SS} = 0 V, T_A = 25°C, 4002B only

SYMBOL	PARAMETER	LIMITS V_{DD} = 5 V MIN	TYP	MAX	V_{DD} = 10 V MIN	TYP	MAX	V_{DD} = 15V MIN	TYP	MAX	UNITS	TEST CONDITIONS See Note 2
t_{PLH}	Propagation Delay		65	110		30	60		20	48	ns	C_L = 50 pF,
t_{PHL}			70	110		30	60		23	48	ns	R_L = 200 kΩ
t_{TLH}	Output Transition Time		75	135		40	70		30	45	ns	Input Transition
t_{THL}			60	135		23	70		15	45	ns	Times < 20 ns

NOTES
1. Additional DC Characteristics are listed in this section under 4000B Series CMOS Family Characteristics.
2. Propagation Delays and Output Transition Times are graphically described in this section under 4000B Series CMOS Family Characteristics

4001B FAIRCHILD 4000B SERIES CMOS FAMILY CHARACTERISTICS

DC CHARACTERISTICS FOR THE 4000B SERIES CMOS FAMILY – Parametric Limits listed below are guaranteed for the entire Fairchild CMOS Family unless otherwise specified on the individual data sheets.

DC CHARACTERISTICS: V_{DD} = 5 V, V_{SS} = 0 V

SYMBOL	PARAMETER			LIMITS MIN	TYP	MAX	UNITS	TEMP	TEST CONDITIONS	
V_{IH}	Input HIGH Voltage			3.5			V	All	Guaranteed Input HIGH Voltage	
V_{IL}	Input LOW Voltage					1.5	V	All	Guaranteed Input Low Voltage	
V_{OH}	Output HIGH Voltage			4.95			V	Min, 25°C	I_{OH} < 1 μA, Inputs at 0 or 5 V per the Logic Function or Truth Table	
				4.95				MAX		
				4.5			V	All	I_{OH} < 1 μA, Inputs at 1.5 or 3.5 V	
V_{OL}	Output LOW Voltage					0.05	V	MIN, 25°C	I_{OL} < 1 μA, Inputs at 0 or 5 V per the Logic Function or Truth Table	
						0.05		MAX		
						0.5	V	All	I_{OL} < 1 μA, Inputs at 1.5 or 3.5 V	
I_{OH}	Output HIGH Current			-0.63			mA	MIN, 25°C	V_{OUT} = 4.6 V	Inputs at 0 or 5 V per the Logic Function or Truth Table
				-0.36				MAX		
I_{OL}	Output LOW Current			1			mA	MIN, 25°C	V_{OUT} = 0.4 V	
				0.8				MAX		
				0.4						
C_{IN}	Input Capacitance Per Unit Load					7.5	pF	25°C	Any Input	
I_{DD}	Quiescent Power Supply Current	Gates	XC			1	μA	MIN, 25°C	All Inputs at 0 V or V_{DD} for all Valid Input Combinations	
						7.5		MAX		
			XM			0.25	μA	MIN, 25°C		
						7.5		MAX		
		Buffers and Flip-Flops	XC			4	μA	MIN, 25°C		
						30		MAX		
			XM			1	μA	MIN, 25°C		
						30		MAX		
		MSI	XC			20	μA	MIN, 25°C		
						150		MAX		
			XM			5	μA	MIN, 25°C		
						150		MAX		

4049B • 4050B

4049B HEX INVERTING BUFFER • 4050B HEX NON-INVERTING BUFFER

DESCRIPTION — These CMOS buffers provide high current output capability suitable for driving TTL or high capacitance loads. Since input voltages in excess of the buffer supply voltage will not be clamped by these buffers, they may also be used to convert logic levels of up to 15 V to standard TTL levels. The 4049B provides six inverting buffers, the 4050B six noninverting buffers. Their guaranteed fan out into common bipolar logic elements is shown in Table 1.

4049B
LOGIC AND CONNECTION DIAGRAM
(DIP (TOP VIEW)

4050B
LOGIC AND CONNECTION DIAGRAM
(DIP (TOP VIEW)

NOTE
Triangle denotes the two power connections diagonally across the package

INPUT PROTECTION
INPUT 200 Ω NOMINAL
TO LOGIC TRANSISTORS
D1
TO V_{SS}
NOTE Typical Breakdown Voltage of Diode D1 is 20 V

TABLE 1
Guaranteed fan out of 4049B, 4050B into common bipolar families

DRIVEN ELEMENT	GUARANTEED FAN OUT
Standard TTL, DTL	2
54LS, 93L, 74LS	1
MSI	16

Conditions: V_{DD} ± 5%, ± 0.25 V
$V_{OL} \leq 0.4$ V, $I_{OL} = 0$ to 6.4 mA

DC CHARACTERISTICS: V_{DD} as shown, V_{SS} = 0 V, 4049BXM and 4050BXM (See Note 1)

SYMBOL	PARAMETER	LIMITS V_{DD} = 5 V MIN	TYP	MAX	V_{DD} = 10 V MIN	TYP	MAX	V_{DD} = 15 V MIN	TYP	MAX	UNITS	TEMP	TEST CONDITIONS
I_{OH}	Output HIGH Current	-1.85 -1.25 -0.9	-2.5								mA	MIN, 25°C MAX	V_{OUT} = 2.5 V for V_{DD} = 5 V Inputs at 0 or V_{DD} per Function
		-0.62 -0.5 -0.35	-1		-1.85 -1.25 -0.9	-2.5		-5.5 -3.75 -2.7	-7.5		mA	MIN, 25°C MAX	V_{OUT} = 4.5 V for V_{DD} = 5 V V_{OUT} = 9.5 V for V_{DD} = 10 V V_{OUT} = 13.5 V for V_{DD} = 15 V Inputs at 0 or V_{DD} per Function
I_{OL}	Output LOW Current	3.75 3 2.1	6		10 8 5.6	16		30 24 16.8	48		mA	MIN, 25°C MAX	V_{OUT} = 0.4 V for V_{DD} = 5 V V_{OUT} = 0.5 V for V_{DD} = 10 V V_{OUT} = 1.5 V for V_{DD} = 15 V Inputs at 0 or V_{DD} per Function
		3.3 2.6 1.8	5.2								mA	MIN, 25°C MAX	V_{OUT} = 0.4 V for V_{DD} = 4.5 V Inputs at 0 V or V_{DD} per Function
I_{DD}	Quiescent Power Supply Current			1 30			2 60			4 120	μA	MIN, 25°C MAX	All inputs at 0 V or V_{DD}

Notes on the following page

DC CHARACTERISTICS: V_{DD} as shown, V_{SS} = 0 V, 4049BXC and 4050BXC (Cont'd) (See Note 1)

SYMBOL	PARAMETER	LIMITS V_{DD} = 5 V MIN	TYP	MAX	V_{DD} = 10 V MIN	TYP	MAX	V_{DD} = 15 V MIN	TYP	MAX	UNITS	TEMP	TEST CONDITIONS
I_{OH}	Output HIGH Current	-1.5 -1.25 -1.0	-2.5								mA mA mA	MIN 25°C MAX	V_{OUT} = 2.5 V for V_{DD} = 5 V Inputs at 0 or V_{DD} per Function
		-0.6 -0.5 -0.4	-1		-1.5 -1.25 -1.0	-2.5		-4.5 -3.75 -3	-7.5		mA mA mA	MIN 25°C MAX	V_{OUT} = 4.5 V for V_{DD} = 5 V V_{OUT} = 9.5 V for V_{DD} = 10 V V_{OUT} = 13.5 V for V_{DD} = 15 V Inputs at 0 or V_{DD} per Function
I_{OL}	Output LOW Current	3.6 3.0 2.5	6		9.6 8 6.6	16		28 24 19	48		mA mA mA	MIN 25°C MAX	V_{OUT} = 0.4 V for V_{DD} = 5 V V_{OUT} = 0.5 V for V_{DD} = 10 V V_{OUT} = 1.5 V for V_{DD} = 15 V Inputs at 0 or V_{DD} per Function
		3.1 2.6 2.1	5.2								mA mA mA	MIN 25°C MAX	V_{OUT} = 0.4 V for V_{DD} = 4.5 V Inputs at 0 V or V_{DD} per Function
I_{DD}	Quiescent Power Supply Current			4 30			8 60			16 120	μA	MIN, 25°C MAX	All inputs at 0 V or V_{DD}

AC CHARACTERISTICS AND SET-UP REQUIREMENTS: V_{DD} as shown, V_{SS} = 0 V, T_A = 25°C, 4049B only (See Note 2)

SYMBOL	PARAMETER	LIMITS V_{DD} = 5 V MIN	TYP	MAX	V_{DD} = 10 V MIN	TYP	MAX	V_{DD} = 15 V MIN	TYP	MAX	UNITS	TEST CONDITIONS
t_{PLH}	Propagation Delay		65	130		30	65		29	52	ns	C_L = 50 pF,
t_{PHL}			50	105		25	50		17	40		R_L = 200 kΩ
t_{TLH}	Output Transition Time		73	145		40	80		30	60	ns	Input Transition
t_{THL}			33	65		13	25		9	20		Times < 20 ns

NOTES:
1. Additional DC Characteristics are listed in this section under 4000B Series CMOS Family Characteristics.
2. Propagation Delays and Output Transition Times are graphically described in this section under 4000B Series CMOS Family Characteristics.

RÉPONSES AUX PROBLÈMES

CHAPITRE UN

1-2. a) 25 b) 9,5625 c) 1241,6875
1-4. 1023
1-5. Neuf bits
1-6. a) quatre circuits pour le mode parallèle b) un circuit pour le mode série

CHAPITRE DEUX

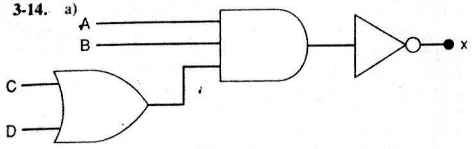
2-1. a) 22 b) 141 c) 2313 d) 983 e) 191
2-2. a) 100101 b) 1110 c) 10111101 d) 11001101 e) 100100001001 f) 111111111
2-3. 255; 65 535
2-4. a) 483 b) 30 c) 2047 d) 175 e) 644
2-5. a) 73_8 b) 564_8 c) 1627_8 d) 200000_8 e) 377_8
2-6. a) 111100011 b) 011110 c) 011111111111 d) 010101111 e) 001010000100
2-7. a) 26_8 b) 215_8 c) 4411_8 d) 1727_8 e) 277_8
2-8. 165, 166, 167, 170, 171, 172, 173, 174, 175, 176, 177, 200
2-9. $100100001001_2 = 4411_8$
2-10. a) 146 b) 422 c) 14 333 d) 704 e) 2047
2-11. a) 4B b) 13A c) 800 d) 6413 e) FFF
2-12. a) 16 b) 8D c) 909 d) 3D7 e) BF
2-13. a) 10010010 b) 000110100110 c) 001101111111101 d) 001011000000 e) 011111111111
2-14. a) 65 536 adresses b) 000 à FFF (hex)
2-15. 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 28A, 28B, 28C, 28D, 28E, 28F, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 29A, 29B, 29C, 29D, 29E, 29F, 2A0
2-16. a) 01000111 b) 100101100010 c) 000110000111 d) 010000100110100010010110001001111
2-17. 10 bits pour le binaire, 12 bits pour le DCB
2-18. a) 9752 b) 184 c) 7775 d) 492
2-19. 11011000(X); 10111101 (=); 10110010 (2); 00110101 (5); 10101111 (/); 01011001 (Y)
2-20. D8, BD, B2, 35, AF, 59
2-21. a) 110110110 (bit de parité dans le rang le plus à gauche) b) 000101000 c) 111110111
2-22. a) 101110100 (bit de parité dans le rang le plus à gauche) b) 000111000 c) 000010100101 d) 11001001000000001
2-23. a) pas d'erreur monobit b) une erreur monobit c) deux erreurs d) pas d'erreur monobit
2-25. a) 10110001001 b) 11111111 c) 209 d) 59 943 e) 4701 f) 777 g) 157 h) 2254 i) 1961 j) 15 900 k) 640 l) 952B m) 100001100101 n) 947 o) 135_{16} p) 5464_8 q) 1001010 r) 01011000 (DCB)
2-26. a) 100101 b) 00110111 c) 01101010 d) 0110011 0110111 e) 45 f) 25
2-27. a) octal b) 16 c) chiffre d) Gray e) Parité; erreurs monobits f) ASCII g) octal; hexadécimal

CHAPITRE TROIS

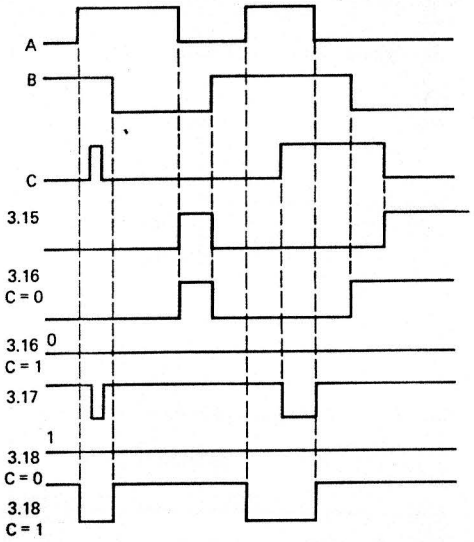
3-1.

3-2.

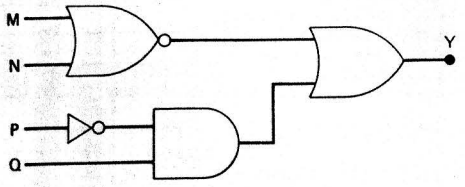
3-3. x est constant au niveau HAUT.
3-5. 31
3-6. a) x est HAUT seulement si A, B, et C sont tous HAUT. b) x est constant au niveau BAS. c) x est HAUT seulement si B et C sont au niveau HAUT simultanément.
3-7. Changer la porte OU pour une porte ET.
3-8. SORTIE est toujours BASSE.
3-10. $x = (\overline{A + B})\overline{BC}$
3-11. $x = \overline{A}\overline{B}\overline{C} + A\overline{B}\overline{C} + \overline{A}\overline{B}D$
3-12. $x = D \cdot (\overline{AB + C}) + E$
3-13. $x = 1$ seulement quand $B = C = 1$ et $A = D = 0$.
3-14. a)



3-15.–3-18.

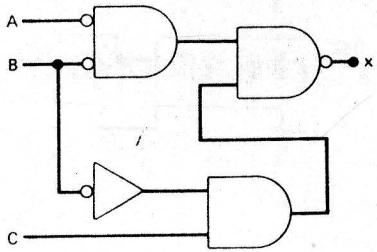


3-19. $x = \overline{(\overline{A + B}) \cdot (\overline{BC})}$
3-20. $x = 0$ seulement quand $A = B = 0$ et $C = 1$.
3-21. c)

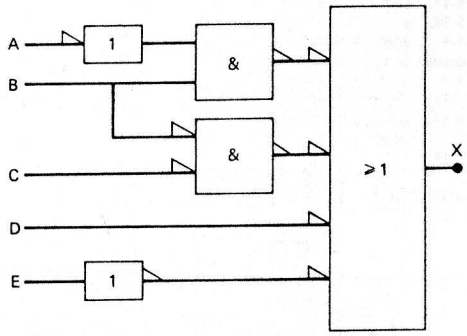


3-22. a) 1 b) A c) 0 d) C e) 0 f) D g) D h) 1 i) G j) y

3-24. a) $MP\overline{N} + \overline{M}\overline{P}N$ b) $B\overline{C}$
3-26. a) $A + \overline{B} + C$ b) $A \cdot (B + \overline{C})$ c) $\overline{A} + \overline{B} + \overline{C}D$ d) $\overline{A} + B + \overline{C} + \overline{D}$ e) $\overline{M}N + M\overline{N}$ f) $(\overline{A} + \overline{B})C + \overline{D}$
3-27. $A + B + \overline{C}$
3-33. a) NI b) ET c) NON-ET
3-34.



3-35. a) Z passe à HAUT seulement quand $A = B = 0$ et $C = D = 1$. b) Z passe à BAS seulement quand A ou B est HAUT ou quand C ou D est BAS.
3-37. X passe à HAUT quand $E = 1$, ou $D = 0$, ou $B = C = 0$, ou quand $B = 1$ et $A = 0$.
3-38. X est BAS pour $EDCBA$ = 01011, 01100, 01101, et 01111.
3-39. a) HAUT b) BAS
3-40. LUMIÈRE = 0 quand $A = B = 0$ ou $A = B = 1$.
3-41.

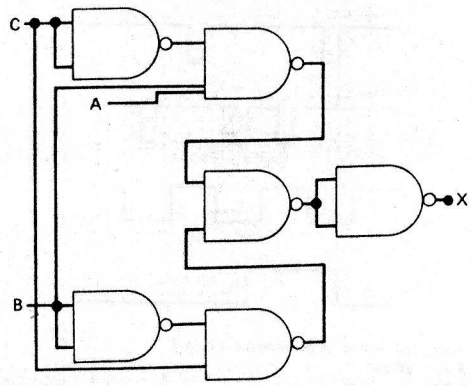


3-42. $Z = \overline{(\overline{\overline{A}\overline{B} + \overline{C}})F + \overline{C} + (\overline{D + E})}$
3-45. Mettre des INVERSEURS sur les entrées A_7, A_5, A_4, A_2 du 74C30.

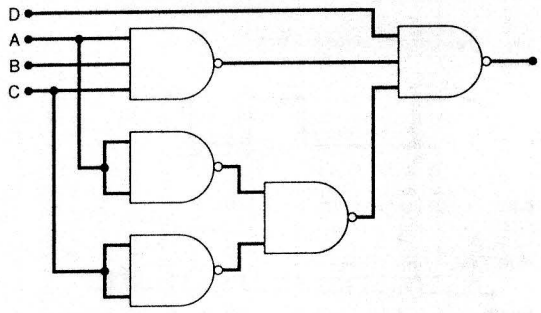
CHAPITRE QUATRE

4-1. a) $C\overline{A} + CB$ b) $\overline{Q}R + Q\overline{R}$ c) $C + \overline{A}$ d) $\overline{R}\overline{S}\overline{T}$ e) $BC + \overline{B}(\overline{C} + A)$ f) $BC + \overline{B}(\overline{C} + A)$ g) $\overline{D} + A\overline{B}\overline{C} + \overline{A}\overline{B}C$
4-2. $Q(M + N)$
4-3. $MN + Q$
4-4. Une solution: $\bar{x} = \overline{B}C + AB\overline{C}$ une autre: $x = \overline{A}B + \overline{B}\overline{C} + B\overline{C}$
4-5. $x = A\overline{B} + \overline{A}\overline{C} + \overline{B}\overline{C}$
4-6. $x = A\overline{B}\overline{C} + ABC$
4-7. $x = \overline{A}_3(A_2 + A_1A_0)$
4-8. Alarme = $ID + \overline{I}L$

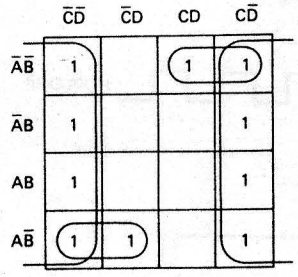
4-9.



4-11.

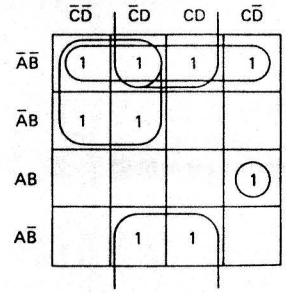


4-12. $x = BC + \overline{B}\overline{C} + AC$; ou $BC + \overline{B}\overline{C} + A\overline{B}$
4-13. $y = \overline{D} + \overline{A}\overline{B}C + A\overline{B}\overline{C}$

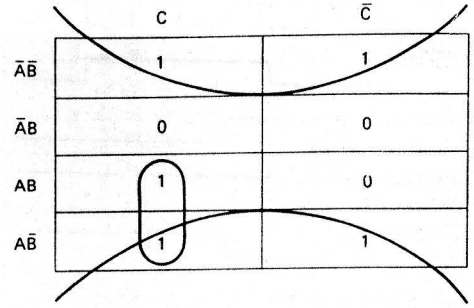


4-14. $x = \overline{A}_3A_2 + \overline{A}_3A_1A_0$

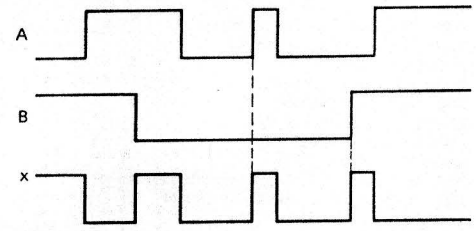
4-15. a) $x = \overline{A}\overline{B} + \overline{A}\overline{C} + \overline{B}D + ABC\overline{D}$



b) $x = \overline{A}\overline{D} + \overline{B}C + \overline{B}\overline{D}$ c) $y = \overline{B} + AC$

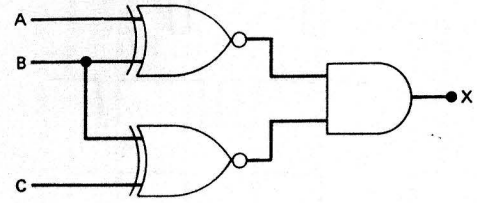


4-16. Meilleure solution: $x = B\overline{C} + AD$
4-17. $x = \overline{S_1S_2} + \overline{S_1S_3} + \overline{S_3S_4} + \overline{S_2S_3} + \overline{S_2S_4}$
4-18. a)

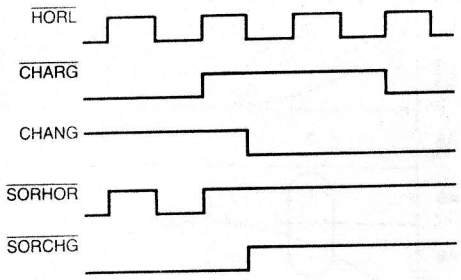


b) $x = A$ c) $x = \overline{A}$

4-19. $A = 0, B = C = 1$
4-20. Une possibilité:



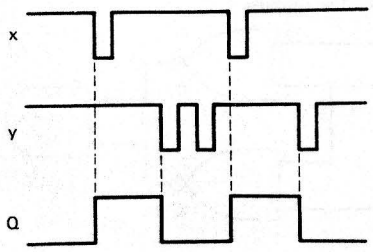
4-22. Quatre sorties où z_3 est le poids fort
$z_3 = y_1y_0x_1y_0$
$z_2 = y_1x_1(\bar{y}_0 + \bar{x}_0)$
$z_1 = y_0x_1(\bar{y}_1 + \bar{x}_0) + y_1x_0(\bar{y}_0 + \bar{x}_1)$
$z_0 = y_0x_0$
4-23. $x = A_3A_2 + A_3A_1$
4-24. $x = AB + (\overline{C \oplus D})$
4-25. $(A + B)(C + D)$
4-26. N-S = $\overline{CD}(A + B) + AB(\overline{C} + \overline{D})$; E-W = $\overline{\text{N-S}}$
4-27. Changer le dernier OU-EX pour un NI-EX
4-29. $x = A\overline{BC}$
4-30. $x = A + BCD$
4-31. $x = A + (B \oplus C)$
4-32. $z = A_0\overline{S} + A_1S$
4-33. $z = x_1x_0y_1y_0 + x_1\bar{x}_0y_1\bar{y}_0 + \bar{x}_1x_0\bar{y}_1y_0 + \bar{x}_1\bar{x}_0\bar{y}_1\bar{y}_0$
Pas de doublet, de quartet ou d'octet
4-34. a) indéterminé b) 1,4–1,8 V c)



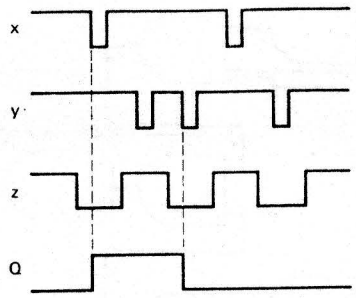
4-38. Défaillances possibles:
V_{CC} ou masse défectueuse dans Z2;
Z2-1 ou Z2-2 ouvert par l'intérieur ou l'extérieur;
Z2-3 ouvert par l'intérieur
4-39. oui: c, e, f non: a, b, d, g
4-41. Z2-6 et Z2-11 en court-circuit
4-43. Défaillances les plus probables:
V_{CC} ou masse défectueuse dans Z1
Z1 enfichée à l'envers
Z1 endommagée à l'intérieur
4-44. Défaillances possibles:
Z2-13 en court-circuit avec V_{CC}
Z2-8 en court-circuit avec V_{CC}
connexion brisée vers Z2-13
Z2-3, Z2-6, Z2-9 ou Z2-10 court-circuités à la masse
4-46. a) 00_{16} à EF_{16} b) F0 c) F8 à FF

CHAPITRE CINQ

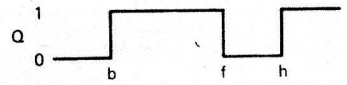
5-1.



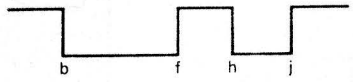
5-2. Même réponse qu'en 5-1
5-3.



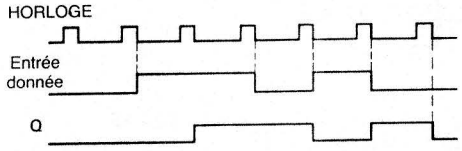
5-6. Z1-4 reste en permanence à HAUT
5-7. 20 ns
5-8.



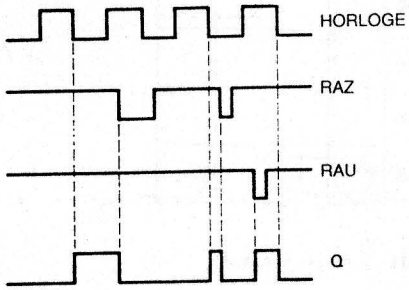
5-9. Q est une onde carrée de 500 Hz
5-10.



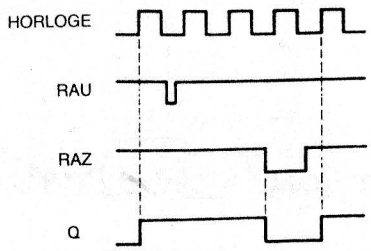
5-11. a) onde carrée de 5 kHz b) 2,5 kHz
5-13.



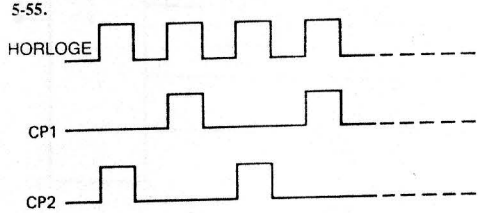
5-14. onde carrée de 500 Hz
5-15. Q demeure BAS
5-18.



5-19.



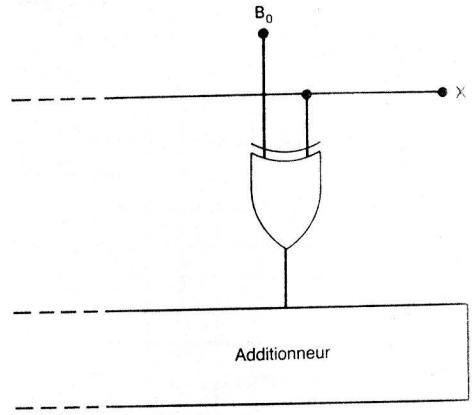
5-20. a) 200 ns b) 7474 c) 30 ns
5-21. 48 ns
5-22. Connecter A à J, $\bar{A}$ à K.
5-23. a) A, B, C
5-24. Connecter X à J, $\bar{X}$ à K
5-25. Utiliser le montage de la figure 5-43
5-26. 1011, 1101, 1110, 0111, 1011, 1101, 1110, 0111, 1011
5-27. Connecter X_0 à l'entrée D de X_3
5-31. a) 10 b) 1953 Hz c) 1024 d) 12
5-32. a) 128 b) 0 à 127
5-35. Placer des INVERSEURS sur A_{11} et A_{14}
5-38. $\overline{Q}$ restera au niveau BAS b) Le FM en $\overline{Q}$ met $X = 1$.
c) fixer $t_p = 20\ \mu s$
5-39. a) A_1 ou A_2 doivent être BAS quand un FM arrive en B.
b) B et A_2 doivent être HAUT quand un FD arrive en A.
5-42. a) 1 V b) 2 V
5-47. 11 s
5-50. a
5-51. Après 14 impulsions, toutes les sorties des circuits cessent de varier avec $A = W = X = Y = Z = 0$ et $B = C = 1$.
5-52. Une possibilité est $R = 1\ k\Omega$ et $C = 80$ nF
5-53. Une possibilité est $R_A = 1\ k\Omega$, $R_B = 10\ k\Omega$, $C = 1800$ pF.
5-55.



5-56. a) non b) oui c) non d) non
5-57. La bascule X_0 a une entrée D ouverte.
5-58. a) oui b) non
5-59. Deux INVERSEURS en cascade entre Q_1 et Q_2 ajoutent suffisamment de retard pour augmenter le t_{PLH} de Q_1, de sorte qu'à la synchronisation de Q_2 le signal Q_1 n'a pas atteint D_2.
5-61. a) non b) non c) oui d) non
5-62. première combinaison: 101 seconde combinaison: 010
5-63. a) non b) non c) oui
5-64. a) mémoire S-C b) JK c) mémoire D d) bascule D
e) bascule et mémoire D f) toutes les bascules
g) SC, JK et D synchrone h) JK

CHAPITRE SIX

6-1. a) 10101 b) 10010 c) 1111,0101 d) 1,1010
e) 100111000
6-2. a) 00100000 (incl. bit de signe) b) 11110010
c) 00111111 d) 10011000 e) 11111111 f) 10000000
g) impossible h) 00000000
6-3. a) +13 b) −3 c) +123 d) −103 e) +127
f) −32 g) −1 h) −127
6-4. a) −2048 à +2047 b) seize bits y compris le bit de signe
6-5. -16_{10} à $+15_{10}$
6-6. a) 164 b) −92 c) DCB invalide d) $ e) −36
6-7. 0 à 1023; −512 à +511
6-9. a) 00001111 b) 11111101 c) 11111011
d) 10000000 e) 00000001 f) 11011110 g) 00000000
h) 00010101
6-11. a) 100011 b) 1111001 c) 100011,00101
d) ,10001111
6-12. a) 11 b) 111 c) 101,11 d) 1111,0011
6-13. a) 10010111 (DCB) b) 10010101 (DCB)
6-14. a) 6E24 b) 100D
6-15. a) OEFE b) 229 c) 02A6
6-16. 16 849 emplacements
6-18. SOMME = $A \oplus B$ REPORT = AB
6-20. [A] = 0000
6-21. 80 ns
6-22. Dépassement = $\bar{A}_3\bar{B}_3S_3 + A_3B_3\bar{S}_3$
6-23. $C_3 = A_2B_2 + (A_2 + B_2)\{A_1B_1 + (A_1 + B_1)[A_0B_0 + A_0C_0 + B_0 + B_0C_0]\}$
6-25. SOMME = 0111 b) SOMME = 1010
6-26.



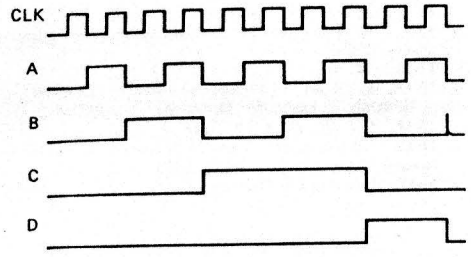
6-27. [S] = 01110; $x = 1$; [Σ] = 0100
6-28. Oui
6-29. [Σ] = 100001000101 (DCB)
6-30. Après la quatrième impulsion d'horloge:
[A] = 00111111; [B] = 01110000 [X] = 0000;
[S] = 10101111
6-31. a) non b) oui c) non
6-34. a) oui b) non c) oui d) non
6-36. Utiliser des bascules D. Connecter $S_3 + S_2 + S_1 + S_0$ à l'entrée D de la bascule zéro; C_4 à l'entrée D de la bascule de report; et S_3 à l'entrée D de la bascule de signe.

CHAPITRE SEPT

7-1. 250 kHz, 50 pour cent
7-2. Même chose que 7-1
7-3. 10000_2

7-4. Il faut cinq bascules: Q_0 à Q_4 où Q_4 est le poids fort. Connecter les sorties Q_3 et Q_4 à une porte NON-ET dont la sortie est raccordée à toutes les bornes RAZ.

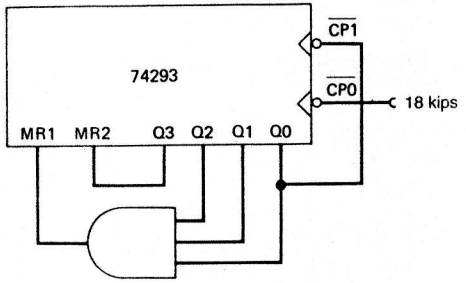
7-5.



7-7. 100, 011, 010, 001, 000, et on recommence

7-8. b) 000, 001, 010, 100, 101, 110, et on recommence

7-9.



7-10. Un MODULO-15 (du problème 7-9) qui pilote un MODULO-4

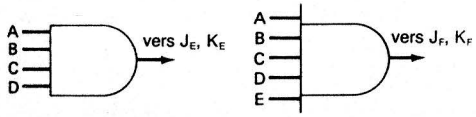
7-11. 60 Hz

7-12. c) 0001

7-13. Les états 1000 et 0000 ne se produisent jamais.

7-14. a) 12,5 MHz b) 8,33 MHz

7-15. a) Ajouter deux bascules (E et F) à la figure 7-60. Connecter des portes ET sous les bascules appropriées.
b) 33 MHz c) 16,7 MHz



7-16. 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, et on recommence

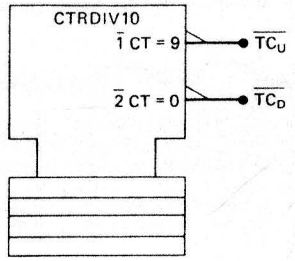
7-18. a) Z n'est pas remis à zéro et la minuterie ne peut être réarmée.

7-19. Change l'entrée de donnée parallèle pour 1010.

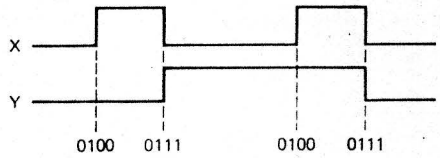
7-20. Neuf

7-21. a) dix b) ramène le compteur à 0000
c) règle le compteur à 1001 d) compteur
e) connecter Q_0 à CP_1 et l'horloge à CP_0; mettre à la masse MR_1, MR_2, MS_1 et MS_2.
f) connecter l'horloge à CP_1; connecter Q_3 à CP_6; mettre à la masse toutes les entrées MR et MS.

7-22. Voici une illustration des changements



7-25.



7-27. a) quand le compteur passe de 0111 à 1000

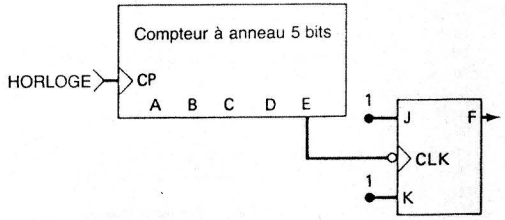
7-28. Douze bascules

7-29. Quatre; seize; treize

7-30. a) $J_A = B\overline{C}$, $K_A = 1$, $J_B = CA + \overline{C}\overline{A}$, $K_B = 1$, $J_C = \overline{A}B$, $K_C = B + A$
b) $J_A = B\overline{C}$, $K_A = 1$, $J_B = K_B = 1$, $J_C = K_C = B$

7-31. $J_A = K_A = 1$, $J_B = K_B = \overline{A}$, $J_C = K_C = \overline{A}\overline{B}$, $J_D = K_D = \overline{A}\overline{B}\overline{C}$

7-33.



Suite du comptage

A	B	C	D	E	F	
1	0	0	0	0	0	←
0	1	0	0	0	0	
0	0	1	0	0	0	
0	0	0	1	0	0	
0	0	0	0	1	0	Répétition
1	0	0	0	0	1	
0	1	0	0	0	1	
0	0	1	0	0	1	
0	0	0	1	0	1	
0	0	0	0	1	1	←

7-34. Suite de comptage: 00000, 10000, 11000, 11100, 11110, 11111, 01111, 00111, 00011, 00001, et on recommence.

7-35. Fréquence à z est 5 Hz

7-36. b) 257 c) 323

7-37. a) 22 b) 450 c) 0 ou 1

7-38. Connecter Q_3 du 74293 à $\overline{CP}_1$ du 74293

7-40. Additionner une seconde bascule J-K

7-42. Connecter Q_3 de l'une à D_S de l'autre

7-43. Le 74178 est garni avec 1101 à l'instant du FD de la sixième impulsion d'horloge.

7-44. 64 μs

7-45. b) $Q_0 \cdot \overline{Q}_1 + Q_0 \cdot Q_7$ c) $Q_3 \cdot \overline{Q}_6$

7-47. a) asynchrone b) vrai c) décalage vers la gauche d) décalage vers la droite e) chargement parallèle f) maintien la donnée g) recirculation par la gauche

7-48. MR_1 sur Q_1 est ouvert, ou Q_3 ou MR_2 est court-circuité à la masse

7-49. RAZ de la bascule X est ouverte ou en permanence à 1 ou ne réagit pas

7-50. Le poids fort du MODULO-6 est en permanence au niveau BAS

7-51. a) Affichage = 001 b) Affichage = 000

7-52. Les compteurs n'ont pas été remis à zéro avant chaque intervalle d'ÉCHANTILLONNAGE

7-53. a) La bascule C bascule à chaque FD de l'entrée b) ne dénombre pas ses états correctement

7-54. P_1 ouvert ou S3 défectueux

7-56. a) parallèle b) binaire c) décompteur d) MODULO-10 e) asynchrone f) en anneau g) Johnson h) tous i) préréglable j) réversible k) asynchrone l) DCB m) synchrone

CHAPITRE HUIT

8-1. a) A, B b) A c) A

8-2. a) 728,3 pJ b) 459,2 pJ c) 58.95 pJ d) 27,41 pJ e) 67,26 pJ

8-3. a) 0,9 V b) 1,4 V

8-4. a) I_{IH} b) I_{CCL} c) t_{PHL} d) produit vitesse-consommation e) V_{NH} f) monté en surface g) consommation de courant h) sortance i) montage en totem j) transistor de rappel à la masse k) 4,75 à 5,25 V l) 2,4 V; 2,0 V m) 0,8 V; 0,4 V

8-5. a) 0,7 V; 0,3 V b) 0,5 V; 0,4 V c) 0,5 V; 0,3 V

8-6. a) 40 μA; 1,6 mA b) ET, NON-ET c) entrées non raccordées

8-7. c) cinq

8-8. a) 30/15 UL b) 24 mA

8-10. Sortance non dépassée

8-11. Sortance non dépassée

8-12. 67 ns

8-13. a) 500 Ω b) 2 kΩ

8-14. b) une résistance de 4,7 kΩ, c'est trop

8-15. a) amplitude trop petite b) t_P trop court c) t_W(L) trop petit

8-18. six portes ET

8-19. AB + CD + FG

8-20. $R_P > 359\ \Omega$

8-21. a) 5 V b) R_S = 110 Ω pour un courant de DEL de 20 mA

8-23. a)

x	y	Donnée sur le bus
0	0	C
0	1	B
1	0	B
1	1	A

8-24. R_P (max) = 5,77 kΩ

8-25. ET

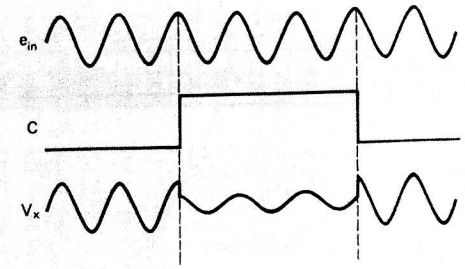
8-29. V_{DD} = 5 V, f_{max} = 10 kHz.

8-30. 3,6 V pour les deux états

8-31. En utilisant les valeurs pour V_{CC} = 6 V, le produit vitesse-consommation est 1, 44 pJ par porte

8-32. 0 V; 1,22 V

8-33.



8-34. −1 et −2

8-35. a) 74HCT b) transpose les tensions logiques c) CMOS ne peut consommer le courant TTL d) faux

8-36. Le choix en c) est une raison possible

8-37. a) aucun b) deux

8-42. Le choix en b) est une défaillance possible

8-43. Une porte NON-ET CMOS est court-circuitée à +5 V

8-44. 0 V à −11.25 V et retour à −6 V

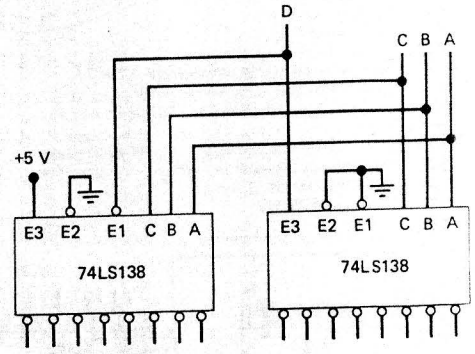
CHAPITRE NEUF

9-1. a) Tous HAUT b) $\overline{O}_0$ = BAS c) $\overline{O}_7$ = BAS d) Tous HAUT

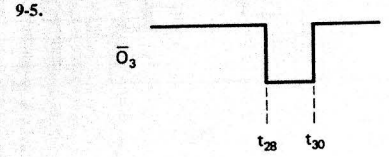
9-2. six entrées, 64 sorties

9-3. [A] = 110, E_3 = 1, $\overline{E}_1 = \overline{E}_2 = 0$

9-4.



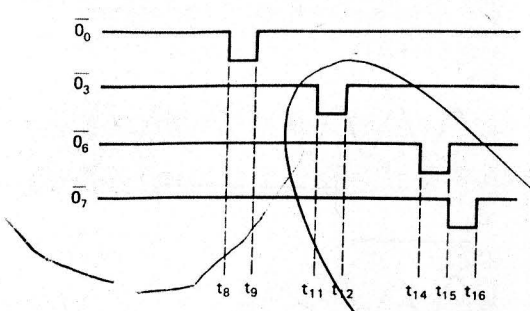
9-5.



9-6. Connecter $\overline{E}_2$ à la masse et prélever la sortie sur $\overline{O}_2$

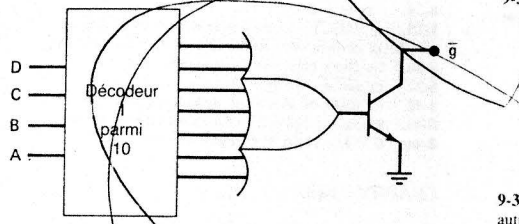
9-7. Validé quand D = 0

9-8.

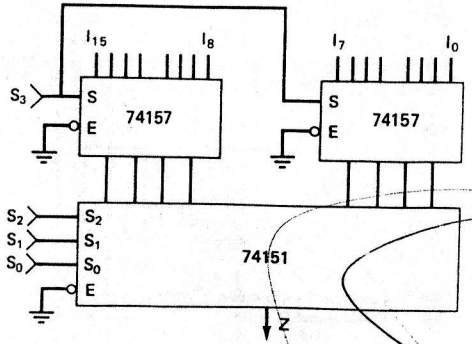


9-10. Les résistances ont 250 Ω

9-12.

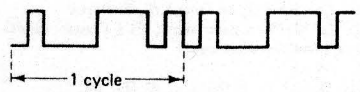


9-13. a) codeur b) codeur c), d) e) décodeur
9-14. 0111
9-17. La quatrième touche enfoncée voit sa valeur placée dans le registre de poids fort
9-18. Choix b)
9-19. Choix a) ou d)
9-20. a) oui b) non c) non
9-21. La ligne de bus A_2 est ouverte entre Z2 et Z3
9-22. E_2 de Z4 est gelé sur le niveau BAS
9-23. Le segment *g* ou le transistor de sortie du décodeur devrait griller
9-24. Les entrées des décodeurs *D* et *C* sont interverties.
9-25. Les sorties du décodeur: *a* et *b* sont en court-circuit
9-27. Un MUX 4 entrées, 1 sortie
9-28. Utiliser neuf 74151.
9-29.

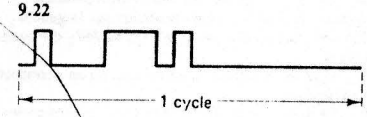


9-30. b) Le nombre total de connexions dans le circuit doté de multiplexeurs est 63, en ne comptant pas V_{CC} et la MASSE, ni les raccordements aux entrées d'horloge du compteur. Le nombre total pour un circuit utilisant des décodeurs/pilotes distincts est 66

9-31.



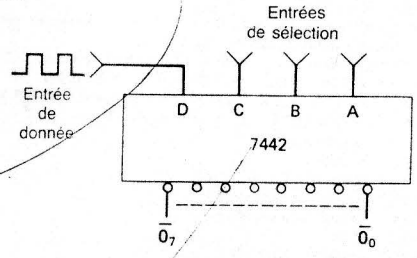
9-32. 9.22



9-33.

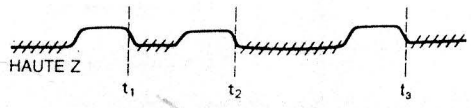
A	B	C	
0	0	0	$0 \rightarrow I_0$
0	0	1	$0 \rightarrow I_1$
0	1	0	$0 \rightarrow I_2$
0	1	1	$1 \rightarrow I_3$
1	0	0	$0 \rightarrow I_4$
1	0	1	$1 \rightarrow I_5$
1	1	0	$1 \rightarrow I_6$
1	1	1	$1 \rightarrow I_7$

9-34. Connecter I_1, I_5, I_9, I_{11}, I_{14}, I_{15} à V_{CC}. Toutes les autres entrées sont à la masse
9-35. *Z* = HAUT pour *DCBA* = 0010, 0100, 1000, 1010.
9-36. b) $Z = \overline{A}BC + A\overline{B}C + AB\overline{C} + \overline{A}\overline{B}\overline{C}D + ABCD$
9-37. a) codeur, MUX b) MUX, DEMUX, c) MUX d) codeur e) codeur, DEMUX f) DEMUX g) MUX
9-38.



9-39. Chaque sortie de DEMUX passe à 0, une à la fois et en séquence.
9-40. Toutes les DEL sont éteintes b) Chaque DEL clignote c) les DEL 2 et 6 clignotent
9-41. Cinq lignes
9-43. a) Le séquencement cesse après la mise en marche du 3e actionneur
b) même qu'en a)
9-44. La défaillance probable est un court-circuit à la masse du poids fort du MUX des dizaines.
9-45. Q_0 et Q_1 sont probablement intervertis
9-46. Les entrées 6 et 7 du MUX sont probablement en court-circuit
9-48. a) non b) oui
9-51. Sortie binaire définitive est 1000101
9-52. A_0 et B_0 probablement intervertis

9-53. $\overline{OE}_C = 0$, $\overline{IE}_C = 1$; $\overline{OE}_B = \overline{OE}_A = 1$; $\overline{IE}_B = \overline{IE}_A = 0$; appliquer une impulsion d'horloge
9-54. a) En t_3, chaque registre renferme 1011
9-55.



9-56. b) ALLUMÉE, ALLUMÉE, ALLUMÉE, ÉTEINTE (gauche à droite)
9-57. a) En t_3, chaque registre contient 1001
9-58. a) 57FA b) 5000 à 57FF c) 9000 à 97FF d) non

CHAPITRE DIX

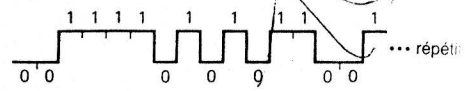
10-1. (f, g) faux
10-2. 3,58 V
10-3. Poids faible = 20 mV
10-4. 20 mV; approx. 0,4 %
10-5. Approx. 5 mV
10-6. Huit
10-7. 14,3 %, 0,286 V
10-9. 250,06 tr/min
10-10. 10 mV, 0,1 %, 6,95 V
10-11.

Binaire	DCB
3,88 mV	10 mV
0,392 %	1,01 %

10-13. 800 ohms; non
10-14. a) −0,3125 V, −4,6875 V b) 4,27 kilohms c) −0,0267
10-15. Utilise moins de valeurs différentes pour *R*
10-16. 20 μA; oui
10-17. a) sept
10-19. 242,5 mV hors norme
10-20. Décalage hors norme
10-21. Le bit 1 du CNA est ouvert ou gelé sur 1
10-22. Les bits 0 et 1 sont intervertis
10-24. a) 10010111 b) 10010111 c) 102 μs, 51 μs
10-27. a) 1,2 mV b) 2,7 mV
10-28. a) 0111110110 b) 0111110111
10-29. 2,834 à 2,844 V
10-31. Rampe numérique a, d, e, f, h CAS b, c, d, e g, h
10-34. 80 μs
10-35. 100101
10-36. 2,96 V
10-37. 2,37 V
10-39. Neuf comparateurs, dix résistances
10-40. Parallèle: a, c, d, f, g CAS: b, f, g Rampe numérique: b, e, f
10-41. a) réversible b) parallèle c) tension-fréquence d) tension-fréquence e) double rampe
10-42. 386 pas
10-43. Le bit de poids fort du chiffre de poids fort n'a pas d'effet sur V_{AX}
10-45. L'interrupteur est bloqué en mode fermeture; l'interrupteur est bloqué en mode ouverture, ou un condensateur est court-circuité
10-47. L'adresse est EAxx

CHAPITRE ONZE

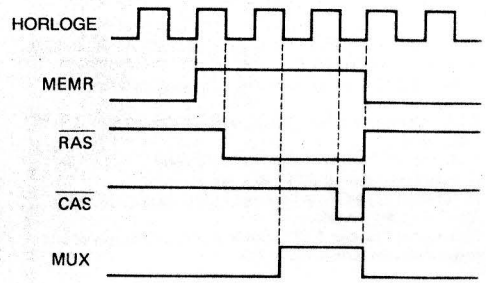
11-1. 16 384; 32; 524 288
11-2. 16 384
11-3. 64K × 4
11-4. 16; 16; 13; 16 384
11-6. a) donnée, adresse, commande b) adresse c) donnée d) UCT
11-7. Haute-Z b) 11101101
11-8. a) registre 11 b) 0100
11-9. a) 16 384 b) quatre c) deux décodeurs 1 parmi 128
11-11. 120 ns
11-12. 180 ns
11-13. a) PROM b) MROM c) EPROM, EEPROM d) EPROM e) EEPROM f) PROM g) EEPROM
11-14. Q_{13}, Q_{14}, Q_{15} à 1, Q_{12} à 0
11-15. Les transistors suivants ont des connexions sur la base ouvertes: Q_0, Q_2, Q_5, Q_6, Q_7, Q_9, Q_{15}.
11-17. La forme d'onde en D_0 est:



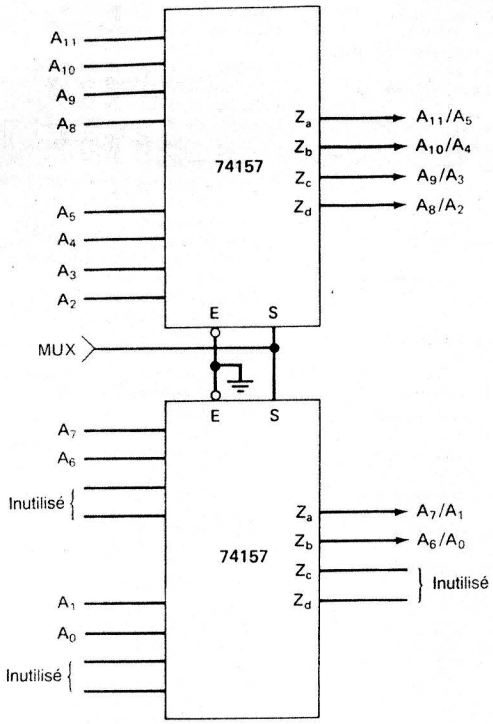
11-18. Données Hexadécimales: 5E, BA, 05, 2F, 99, FB, 00, ED, 3C, FF, B8, C7, 27, EA, 52, 5B
11-19.

A_5	A_4	A_3	A_2	A_1	A_0	D_2	D_1	D_0	C_3	C_2	C_1	C_0
1	0	0	1	0	1	0	1	1	0	1	1	1
0	0	1	1	1	0	0	0	1	0	1	0	0
1	1	1	1	1	0	1	1	0	0	0	1	0
0	1	1	1	0	0	0	1	0	0	1	1	0
1	1	0	1	0	1	1	0	1	0	0	1	1
0	0	0	1	1	0	0	0	0	0	1	1	0
0	1	0	0	1	1	0	0	1	1	0	0	1
1	1	0	0	0	0	1	0	0	1	0	0	0

11-20. $O_3 = D$; $O_2 = C\overline{D} + \overline{C}D$; $O_1 = B\overline{C} + \overline{B}C$; $O_0 = A\overline{B} + \overline{A}B$
11-21.
$O_3 = D + 0 + 0 + 0$
$O_2 = C\overline{D} + \overline{C}D + 0 + 0$
$O_1 = B\overline{C} + \overline{B}C + 0 + 0$
$O_0 = A\overline{B} + \overline{A}B + 0 + 0$
11-23. a) 100 ns b) 30 ns c) dix millions d) 20 ns e) 30 ns f) 40 ns g) dix millions
11-24. Sept
11-25.



11-26.



11-29. Le MUX de la figure 11-55 est interverti et connecté à la VALIDATION RANGÉE de la figure 11-35

11-32. Ajouter quatre PROM de plus (PROM-4 à PROM-7) au circuit. Connecter leurs sorties de donnée et leurs entrées d'adresse aux bus de donnée et d'adresse, respectivement. Connecter AB_{13} à l'entrée *C* du décodeur et raccorder les sorties 4 à 7 du décodeur aux entrées *CS* des PROM 4 à 7, respectivement.

11-33. Connecter AB_{13}, AB_{14} et AB_{15} à une porte OU 3 entrées et connecter la sortie de cette porte OU à l'entrée *C* du décodeur.

11-34. a) 32 × 8 b) VIVES 2 et 3 c) 00-0F; 10-1F

11-37. *PEACE

11-39. F000-F3FF; F400-F7FF; F800-FBFF; FC00-FFFF

11-41. L'entrée *B* du décodeur est ouverte ou bloquée au niveau HAUT

11-42. Le noeud sortie de la porte OU/entrée *C* est bloqué à zéro

11-43. Seuls les modules de VIVES 1 et 3 sont testés

11-44. La puce VIVE ayant les sorties de donnée 4 à 7 affiche une défectuosité sur chaque bit

11-45. La sortie 7 du module VIVE 3 est ouverte ou bloquée à 1

11-47. Somme de contrôle = 11101010

INDEX

A

Acheminement des données, par MUX, 502-503
ACL (*voyez* Afficheurs à cristaux liquides)
Acquisition de données, 580-583
Action à consommation du courant, TTL, 395
Action à injection de courant, 395-396
Actionneur, 556
ADC0804, CAN par approximations successives 8 bits, 586-589
Addition
  binaire, 246
  DCB, 257-259
  hexadécimale, 259-260
  logique (OU), 43, 44-48, 246
  système complément à 2, 253-254, 273-276
Addition binaire, 246
  comparaison avec l'addition logique, 246
Additionneur binaire parallèle, 263-264
  circuits intégrés, 271-273
  complet, avec registres, 267-269
  étude d'un cas de dépannage, 285-287
  notation en complément à 2, 273-276
  propagation du report, 269-271
Additionneur complet, 264
  conception d'un, 264-267
  simplification avec un diagramme K, 267
Additionneur parallèle intégré, 271-273
Additionneurs
  DCB, 276-280
  complet, 264
  parallèle, 264
  parallèle binaire (*voyez* additionneurs parallèles binaires)
Additionneurs DCB, 276-280
Additionneurs DCB en cascade, 278
Additionneurs parallèles en cascade, 271-272
Adresse, 616, 710, 717-718
Adresse d'opérande, 711, 717
Afficheurs
  ACL, 485-487
  DEL, 483-485
  incandescents à 7 segments, 485
Afficheurs à cristaux liquides (ACL), 485-487
  attaque, 485-487
  Plaque arrière, 485
Algèbre booléenne, 41-85
  constantes et variables, 42-43
  description des circuits logiques, 52-54
  évaluation des sorties des circuits logiques, 54-56
  matérialisation des circuits à partir d'expressions, 56-57
  opération ET, 48-49
  opération NON, 43, 51-52
  opération OU, 43, 44-48
  porte NI, 58-59, 69-71
  porte NON-ET, 59-60, 69-72
  récapitulation, 45-46
  représentation synonyme des portes logiques, 72-76
  sélection de la représentation des portes logiques, 76-82
  simplification des circuits logiques, 95-98
  tables de vérité, 43-44
  théorèmes, 61-65
  théorèmes de De Morgan, 65-68
American Standard Code for Information Interchange (ASCII), 33-34
  liste partielle du, 33
Amplificateur de détection (dans une DRAM), 659
Amplificateur opérationnel (dans CNA), 566-570
Arithmétique numérique, 244-287
  addition binaire, 246
  addition DCB, 257-259
  addition hexadécimale, 259-260
  additionneur binaire parallèle, 263-264
  additionneur complet, 263
  additionneur DCB, 276-280
  additionneur parallèle intégré, 271-273
  circuits arithmétiques, 262
  circuits et fonctionnement, 244-287
  division binaire, 257
  fonctionnement et circuits, 244-287
  multiplicateurs binaires, 280-284
  multiplication binaire, 256
  notation complément à 2, multiplication, 256
  notation complément à 2, addition, 253-254, 273
  notation complément à 2, soustraction, 254-255
  propagation du report, 270-271
  représentation de nombres signés, 247-272
  soustraction hexadécimale, 260-262
Autotest au démarrage, VIVE, 691

B

/ (*voyez* Notation fonctionnelle)
Bascules, 15, 154-228
  applications, 191
  circuits séquentiels, analyse, 218-220
  considérations sur la synchronisation, 186-189
  définition, 156
  dépannage des circuits, 222-227

détection d'une séquence d'entrée, 193
dispositifs connexes, 154-228
division de fréquence et comptage, 199-203
données, 176-178
  réalisation, 177
élément de mémoire à portes NON-ET, 157-161
  étude d'un cas de dépannage, 166
  représentation synonyme, 160-161
élément de mémoire à portes NI, 163-165
élément de mémoire D, 178-180
éléments de mémoire, 14
entrées asynchrones, 180-183
entrées prioritaires, 181
état au démarrage, 165
maître/esclave, 191
NMOS, 434
problèmes de synchronisation, 189-191
récapitulation, 227-228
registres à décalage, 196-199
signaux d'horloge, 167-169
synchronisation, 191-193
synchronisées D, 176-178
  réalisation, 177
synchronisées J-K, 173-176
synchronisées, principes des, 167-169
synchronisées S-C, 170-173
temps d'établissement et de maintien, 169
Bascules déclenchées, 167-169
D synchronisée, 176-178
élément de mémoire D, 178-180
entrées asynchrones, 180-183
J-K synchronisée, 173-176
principes, 168-169
S-C synchronisée, 170-173
Bascules maître/esclave, 191
avec verrouillage de la donnée, 191
BASIC, langage, 721
BCD/DCE (*voyez* notation fonctionnelle)
BIN/OCT (*voyez* notation fonctionnelle)
Bit, 9
Bit de dépassement, 267
Bit de poids le plus fort (MSB), 8-9
Bit de poids le plus faible (LSB), 8-10
Bit de report, 246, 263
Bit de signe, 247, 254-256
Bit de somme, 263
Bit unité, 10
Bits de grandeur, 247
Bloc de contrôle commun, 185, 307 (*voyez* Symboles logiques standard IEEE/ANSI)
Boîtier à double rangée en ligne, 23
Boîtier plat en céramique, 392
Branchement conditionnel, 710
Broches entrées/sorties communes (dans les VIVES), 652
Bruit, 5, 222
Bruit électrique, 35
Bus bidirectionnels, 536-537, 621
Bus d'adresse, 623, 661-662, 668, 725
Bus de commande, 623
Bus de donnée, 528-537, 623, 726
Bus, extension, 534-535
Bus unidirectionnel, 725

C

C (*voyez* notation fonctionnelle)
CI (*voyez* Notation fonctionnelle)
CAN (*voyez* Convertisseur analogique-numérique)
CAN à poursuite, 591-592
CAN double pente, 592
CAN flash (parallèle), 589-591
  durée de conversion, 591
CAN par approximations successives, 583-589
CAN rampe numérique, 576-580
  bidirectionnel, 591-592
CAN rampe numérique bidirectionnel, 591-592
CAN tension-fréquence, 592
Capacité, mémoire
  définition, 615
  extension, 670-674
Cellule mémoire, 615
Cellule mémoire NMOS, 653
Charge des TTL, 405-411
Charge unitaire (U.L.), 408
Chiffre binaire, 9
Chiffre de poids le plus faible (LSD), 7
Chiffre de poids le plus fort (MSD), 7
Chiffres, 3, 7
Chronogrammes, 14
CI numérique unipolaire, 130 (*voyez* Famille logique CMOS)
CI numériques bipolaires, 130-132
  ECL, 426-429 (*voyez* Famille logique TTL)
Circuit à sortie en totem, 396, 416
Circuit d'aiguillage d'impulsions, 129, 172-173
Circuit détecteur de front, 172-173
Circuit NI exclusif, 121-122
Circuit NON (INVERSEUR), 51-52
  choix de la représentation, 76-82 (*voyez* Circuits logiques combinatoires)
  CMOS, 436-437
  définition, 51-52
  description booléenne, 53
  évaluation de la sortie, 54-56
  matérialisation des expressions booléennes, 56-57
  NMOS, 432-432
  représentation synonyme, 72-82
  théorème de De Morgan, 65-68
Circuit OU exclusif, 120
Circuits arithmétiques, 262-262
Circuits de blocage, 127-129
Circuits échantillonnage et mémorisation, 596-597
Circuits intégrés (*voyez* Familles logiques de circuits intégrés)
Circuits intégrés numériques, 12-13
Circuits logiques
  analyse, 78-79, 218-220
  arithmétiques, 262-287
  combinatoires (*voyez* Circuit logiques combinatoires)
  définition, 12
  détermination du niveau de sortie d'après un diagramme, 55-56
  éléments de mémoires (*voyez* Bascules)
  portes logiques (*voyez* Portes logiques)
  schéma de connexion (logigramme), 133-134
Circuits logiques combinatoires, 93-143
  conception, 98-108
  forme somme de produits, 95
  générateur et contrôleur de parité, 121-124
  matérialisation, 105-107
  NI exclusif, 121-124
  OU exclusif, 119-120
  procédure de conception complète, 102
  simplification, 95-98
  simplification algébrique, 96-98
  simplification avec diagrammes de Karnaugh, 108-119, 266-267
  sorties multiples, 128
Circuits logiques MSI, 451-537
  afficheurs à cristaux liquides (ACL), 485-487
  bus de donnée, 530-537
  codeurs, 487-493
  DCB-décimal, 480-481
  décodeurs, 475-486
  démultiplexeurs (DEMUX), 508-516
  multiplexeurs (MUX), 496-502
  pilotes DCB-7 segments, 483-485
  registres trois états, 528-530
Circuits numériques, 11-13 (*voyez* Circuits logiques)
Circuits séquentiels
  analyse, 218-220
  conception, 334-340
CNA (*voyez* Convertisseur numérique-analogique)
CNA à réseau R/2R, 569-570
CNA bipolaire, 565
Codage, 29
Codage en binaire pur, 29
Code ASCII, 33-34
  liste partielle du, 33
Code d'adresse, 203
Code d'entrée DCB d'un CNA, 563
Code d'opération, 717
Code DCB (décimal codé binaire), 29-31
  avantages, 30
  comparaison avec le code binaire, 30
Code Gray, 32
Code majoré de trois, 31
Code non pondéré, 32
Codes (*voyez* Codes binaires)
Codes à distance minimale, 32
Codes alphanumériques, 33-34
Codes binaires, 29-34
  alphanumérique, 33-34
  DCB, 29-31
  Gray, 32
  majoré de 3, 31
  méthode de la parité pour la détection des erreurs
Codeur 8 voies d'entrée, 3 voies de sortie, 487-488
Codeur de priorité décimal-DCB (74347), 488-493
Codeurs, 487-493
  8 voies d'entrée, 3 voies de sortie, 487-488
  d'interrupteurs, 490-493
  de priorité, 488-493
  de priorité décimal-DCB, 488-493
  octal-binaire, 487-488
Codeurs d'interrupteurs, 490-493
Codeurs de priorité, 488-493
Codeurs octal-binaire, 487-488
Comparateur de grandeurs, 521-524
  applications, 523
  entrées des données, 521
  entrées en cascade, 521-522
  sorties, 521
  symbole IEEE/ANSI, 524
Complément (*voyez* Inverseur)
Complément à 2, cas spécial d'une représentation, 251-252
Complémentation ou inversion logique (opération NON), 43, 51-52
Complexité des circuits intégrés, 392
Comptage
  binaire, 9-10
  décimal, 7-8
  hexadécimal, 28
  octal, 24
  opération de , 200-201
Compteur asynchrone CMOS, 308
Compteur binaire, 200-201, 513-514
Compteur de mots, 513-514
Compteur de rafraîchissement, 666
Compteur en anneau, 340-341, 491-492
  diagramme d'états d'un, 341
  en circuit, 492
  lancement d'un, 342
Compteur Johnson, 342-344
  décodage, 343-344
Compteurs, 294-366
  4731B, 357
  74164, 358
  74165, 358
  74174 353
  74178, 353
  74193 (LS193/HC193), 319-324
  74293, 303-308
  7493, 303-308
  74HC160, 315
  74HC161, 315
  74HC162, 315
  74HC163, 315
  74HC164, 358
  74HC165, 358
  74HC4024, 308
  74HC4040, 308
  74LS160, 315
  74LS161, 315
  74LS162, 315
  74LS163, 315
  74LS164, 358
  74LS165, 358
  à anneau, 340-342, 491-492
  à décade, 286-303
  à propagation, 296-313
  applications, 294-366
  asynchrones (à propagation), 296-313
  asynchrones (à propagation) décompteur, 309-310
  asynchrones (à propagation) retard de propagation, 310-313
  asynchrones intégrés, 303-308
  bit, 513
  conception, synchrone, 334-340
  DCB, décodage, 330
  DCB en cascade, 333-334
  décodage, 326-330, 343
  dépannage, 363-366
  fréquence, 344-349
  horloge numérique, 350-352
  Johnson, 342-344
  MODULO < 2^N, avec, 299-303
  mot, 513
  pointes, 300
  pointes de décodage, 330-333
  préréglable, 317-319
  recyclage, 297
  registre à décalage, 340-344
  registre lecture parallèle/écriture série, 358
  registre lecture parallèle/écriture parallèle, 353-357
  registre lecture série/écriture série, 357
  registre lecture série/écriture parallèle, 358
  registres intégrés, 353-362
  rétrocouplage, avec, 340
  synchrone parallèle, 313-324
  synchrone parallèle compteur/décompteur, 315-317
Compteurs à décade, 190-191
Compteurs à décalage intégrés, 344
Compteurs à propagation (*voyez* Compteurs asynchrones)
Compteurs à registre à décalage, 340-344
Compteurs asynchrones (à propagation), 296-313
  circuit intégré, 303-308
  décompteur, 308-310
  numéro MODULO, 297-303
  retard de propagation, 310-313
Compteurs circulaires rétrocouplés, 342-344
Compteurs DCB, 286-303
  décodage, 330
  montés en cascade, 333-334
Compteurs DCB en cascade, 333-334
Compteurs de fréquence, 344-3449
Compteurs parallèles (*voyez* Compteurs parallèles synchrones)
Compteurs préréglables, 317-319
Compteurs synchrones (parallèles), 313-324
  avantages sur les asynchrones, 314
  compteur/décompteur (réversible), 314-317
  conception, 334-340
  préréglables, 317-319
  74193 (LS193/HC193), 319-324
Conditions « indifférentes », 118-119
Conducteur fusible, PROM, 632
Conducteurs de donnée, 203
Conflit de bus, 425
Conformateur d'impulsion, 298
Connexions de la mémoire, UCT, 621-623
Conséquences des théorèmes de De Morgan, 67-68
Contrôleur de parité, 125-126
Conversion analogique-numérique (*voyez* Convertisseur analogique-numérique)
Conversion numérique-analogique (*voyez* Convertisseur numérique-analogique)
Conversion parallèle-série, 504-505
Convertisseur analogique-numérique (CAN), 6, 555, 574-599
  acquisition des données, 580-583
  circuit échantillonnage et mémorisation, 596-597
  double pente, 592
  durée de conversion, 579-580, 585
  erreur de quantification, 578
  flash, 589-591,
  multiplexage, 597-599
  par approximations successives, 583-589 (*voyez aussi* Convertisseur numérique-analogique)
  poursuite, 591
  précision, 578-579
  puce, par approximations successives 8 bits, 586-589
  rampe numérique, 576-580
  rampe numérique bidirectionnel, 591
  résolution, 578-579
  tension-fréquence, 592
  voltmètre numérique, 593-596
Convertisseur de données, 639
Convertisseur numérique-analogique (CNA), 6, 556-574
  à réseau R/2R, 569-570
  applications, 572-573
  bipolaire, 564
  circuits, 564-570
  code d'entrée DCB, 563-564
  commande, utilisée dans le, 573
  conversion analogique-numérique, utilisé dans la, 573
  dépannage, 573-574
  erreur de décalage, 572
  escalier, 559
  exactitude, 567-568
  forme d'onde de sortie, 559
  hauteur de l'échelon, 559
  monotonicité, 572
  pondération des entrées, 558-559
  reconstruction du signal, utilisé dans la, 573
  résolution, 559-562
  résolution en pourcentage, 561
  sortie analogique, 558
  sortie de courant, avec, 568-569
  sortie pleine échelle, 559
  spécifications, 570-572
  temps d'établissement, 571-572
  tension de décalage, 572
  test d'exactitude statique, 573-574
  test en escalier, 573-574

Convertisseurs de code, 524-528
autres réalisations, 528
matérialisation des circuits, 526-527
principe de base, 525
procédé de conversion, 525-526
CT (*voyez* Notation fonctionnelle)
CTR (*voyez* Notation fonctionnelle)
Cumulande, 253
Cumulateur, 253
Cycle d'exécution, 723
Cycle de prise en charge, 723

D

D (*voyez* Notation fonctionnelle)
DBS, 401
Décharge électrostatique, 435
Décimal codé binaire (*voyez* Code DCB)
Déclenché par un front, 168
Décodage
compteur Johnson, 343
compteurs, 326-330
pointes, 330-333
Décodage vrai au niveau HAUT, 327-328
Décodage vrai au niveau BAS, 328
Décodeur 3 voies d'entrée, 8 voies de sortie, 475-476
Décodeur DCB-décimal, 480-481
Décodeur/pilote DCB-7 segments, 483-485
Décodeur/pilote DCB-décimal, 481
Décodeurs, 475-487
1 parmi 10, 480-481
1 parmi 8 (*voyez* binaire-octal)
3 voies d'entrée, 8 voies de sortie, 475-476
4 parmi 10 (*voyez* DCB-décimal)
adresse, 625
afficheurs à cristaux liquides (ACL), 485-487
applications, 481
binaire-octal, 475-476
DCB-décimal, 480-481
/démultiplexeur, 508
entrées de VALIDATION, 476
partage de temps, 503
pilotes DCB-7 segments, 483-485
Décodeurs d'adresse, MORTE, 625
Décompteur (*voyez* Compteurs)
Découplage, alimentation TTL, 415
Découplage de l'alimentation, TTL, 415
DEL (*voyez* Diode électroluminescente)
Demi-additionneur, 267
Systèmes à disque dur, 682
chargeurs de disque, 682
disque dur, 682
écrasement de têtes, 683
Démultiplexeurs (DEMUX), 508-516
1 voie d'entrée, 8 voies de sortie, 508-513
horloge, 510-511
système de télésurveillance de sécurité, 510-512
DEMUX (*voyez* Démultiplexeurs)
Dépannage (*voyez* Systèmes numériques, dépannage)
additionneur binaire parallèle, 285-287
circuit d'affichage d'un compteur DCB multichiffres, 518-519
circuits bascules, 222-227
sorties court-circuitées, 224
entrées ouvertes, 222-227
compteurs, 363-366
convertisseurs numériques-analogiques, 573-574
décodeurs, circuits avec, 495-496
défaillances externes des CI, 139-141
défaillances internes des CI, 135-139
dépistage des noeuds court-circuités, 457
étapes de base, 134-135, 456-458
étude de cas
additionneur binaire parallèle, 285-287
portes, 141-143
observation/analyse, 495-496
outils servant en, 135, 143, 456-458
système de télésurveillance de sécurité, 519-520
système de transmission de données synchrone, 520-521
systèmes de mémoire VIVE, 684-692
systèmes numériques, 134-143 (*voyez* Systèmes numériques)
Dépassement arithmétique, 256
Destructive, 681
Détection d'une séquence d'entrée, 193
Détection des erreurs, méthode de la parité, 35-37
Diagramme de Karnaugh
format, 109-110
méthode, 109-119
processus de simplification complet, 113
récapitulation, 119
simplification 266-267
Diminuande, 254
Diminuteur, 254
Diode à barrière de Schottky (DBS), 401
Diodes électroluminescentes, 483-485
anode commune, 483
cathode commune, 483
DIP (boîtier double rangée en ligne), 129
Dispositifs à bascule de Schmitt, ***204-206***
Dispositifs mémoires, 612-693
dynamiques, 617
statiques, 617
Disque
chargeurs, 682
Disquette, 683
Disques durs, 682
Disques optiques, 684
disque compact ROM (CD-ROM), 684
disque WORM (inscriptible une seule fois), 684
lecture/écriture, 684
ROM optique, 684
Distributeurs de données, 508-516
Dividende, 257
Diviseur, 257
Diviseur de fréquence variable, 324
Division binaire, 257
Division de la fréquence, 199-200, 298
DIV*n* (*voyez* Notation fonctionnelle)
DRAM (*voyez* Mémoire VIVE dynamique)
Durée de conversion, CAN, 579-580, 585
Durée de l'impulsion, 390

E

Échanges avec le monde extérieur, 553-600
ECL, famille de circuits intégrés, 426-429 (*voyez* Logique à couplage par l'émetteur)
Écrasement de têtes, 683
EEPROM (PROM effaçable électriquement), 638-638
Élément de mémoire D, 178-180 (*voyez* Bascules)
Élément PAL, 643-644 (*voyez* Éléments logiques programmables)
Élément PLA, 644-645 (*voyez* Éléments logiques programmables)
élément FPLA, 644
polarité de sortie programmable, 645
Éléments de mémoire, 15, 157-165, 178-180 (*voyez* Bascules)
Éléments logiques programmables, 641-649
architecture des PROM, 643
autres fonctions, 647-648
cartes des conducteurs fusibles, 648
effaçable, 648-649
élément PAL, 643-644
principe de base, 641-613
programmation, 648
symbologie, 613-643
Entrée d'horloge de comptage, 319
Entrée d'horloge de décomptage, 319
Entrée de préréglage, 319
Entrée de VALIDATION, décodeurs, 476
Entrée lecture/écriture (L/E ou R/W), 619
Entrées asynchrones, 180-183
désignations des, 181-182
Entrées d'adresse, 496, 619
Entrées de commande, 168-169, 180
synchrones, 168
Entrées de commande synchrones 168
Entrées de déclenchement, 171
Entrées de sélection (MUX),496
Entrées de strobe (dans un DRAM), 661
Entrées flottantes, 133 411 (*voyez* entrées non connectées)
Entrées inutilisées
CMOS, 133, 443
TTL, 133, 411
Entrées non connectées
CMOS, 133, 443
TTL, 133, 411
Entrées réunies ensemble, TTL, 412
Entrées synchrones, 168, 180
EPROM (MORTES programmables effaçables), 633-638
Erreur de décalage, 572
Erreur de linéarité (d'un CNA), 571
Erreur de quantification, 579
Erreur pleine échelle (d'un CNA), 571
État à un, 159
État invalidé (Haute-Z), 424
État quasi-stable, 206
État RAZ, 159
État remis à zéro, 159
État validé, 424
Exécution d'un programme en langage machine, 720-724
Exigences d'alimentation, circuit intégré, 387

F

Facteur de charge, 386
Famille logique CMOS, 12, 131-134, 385, 436-458
bascule S-C, 438
caractéristiques des séries, 439-444
consommation, 441
entrées inutilisées, 443
entrées non connectées, 133
gammes des tensions des niveaux logiques, 132
INVERSEUR, 130, 436-437
marges de sensibilité au bruit, 441
masse, 132
niveaux de tension, 440
pilotant un TTL, 452-456
piloté par un TTL, 429-452
porte de transmission, 447-449
porte NI, 438
porte NON-ET, 437-438
série 4000/14000, 131-132
série 74C, 131-132, 440
série 74HC, 131-132, 440
série 74HCT, 131-132, 440
sortance, 442
sorties à drain ouvert, 444-446
sorties court-circuitées, 444-446
sorties trois états, 447-449
susceptibilité à l'électrostatique, 443
tension d'alimentation, 132, 439
vitesse de commutation, 443
Famille logique MOS, 430-458
caractéristiques, 434-435
complémentaire (*voyez* Famille logique CMOS)
complexité du procédé, 435
décharge électrostatique, 435
NMOS, 432-434 (*voyez* Mémoire)
vitesse de fonctionnement, 434
sensibilité à l'électrostatique, 435
sortance, 434
transistor MOS, 430-431
Famille logique NMOS, 12
circuits logiques, 432-434
marges de sensibilité aux bruits, 434
vitesse de fonctionnement, 434
Famille logique TTL, 131-133
action de consommation de courant, TTL, 395
caractéristiques standard des séries, 397-400, 404
circuit à sortie en totem, 392 396-397
consommation, 399
courants transitoires, TTL, 414-415
définition, 12
entrées inutilisées, 411-412
entrées non connectées, 133, 411
entrées réunies ensemble, 412
faible consommation Schottky, série 74LS, 402-403
faible consommation, série 74L, 400-401, 404
fiches techniques des fabricants, 397
intervalle de températures, 397
mise à 0 des entrées, 413
mise à la masse, 132-133
niveaux de tension, 399
opération des circuits – état BAS, 394-395
opération des circuits – état HAUT, 395
pilotant un 74HCT, 451
pilotant un CMOS, 451-452
pilotant un CMOS haute tension, 451-452
piloté par un CMOS, 452-456
porte NON-ET, de base, 392
Schottky, série 74S, 401-402, 404
sortance, 399, 405-411
sortie à collecteur ouvert, 415-422
sous-familles, 131, 400-404
tension d'alimentation, 132, 133
trois états, 422-426
valeurs nominales maximales pour la tension, 399
Familles logiques de circuits intégrés, 383-458
caractéristiques de base, 129-134
bipolaires, 130-1298, 426
définition, 384-385
ECL, 426-429
interfaçage, 449-456
MOS (*voyez* Famille logique MOS)
terminologie, 385-392
TTL (*voyez* Famille logique TTL)
unipolaires, 130-132
FILE (*voyez* Mémoire FILE)
vin
→ (*voyez* Notation fonctionnelle)
Fonction de commande, 325
Fonction ET, 325-326
Forme d'onde en escalier, d'un CNA, 559
Forme somme de produits, 95
Fréquence de synchronisation maximale (f_{MAX}), 187
Front montant (FM), 168
Fronts, d'un signal d'horloge, 167-168

G

G (*voyez* Notation fonctionnelle)
Générateur de fonctions, 640
Générateurs de caractères, 640
Génération de fonctions logiques, 505-507
Grand ordinateur, 16-17
Grandeurs binaires, représentation des, 10-11

H

Hauteur du pas, 559 (*voyez* Résolution; Convertisseur numérique-analogique)
Horloge
démultiplexeur, 510-511
dérive, 225-226
durées de transition, 187-188
impulsion BASSE t_W(L), 187
impulsion HAUTE t_W(H), 187
signaux, 167-169, 726
Horloge numérique, 350-352
HPRI/BCD (*voyez* Notation fonctionnelle)

I

Immunité aux bruits, 388-389
Instruction de décision (*voyez* Branchement conditionnel)
Instructions à adresses multiples, 718
Instructions multioctets,719-720
Intégration à grande échelle (LSI), 129, 707
Intégration à moyenne échelle (MSI), 129
Intégration à petite échelle (SSI), 129
Intégration à très grande échelle (VLSI), 129
Intégration à ultragrande échelle (ULSI), 129
Interfaçage
circuit intégré, 449-456
dans l'ordinateur, 712-713
numérique/analogique (*voyez* Convertisseur analogique-numérique; convertisseur numérique-analogique)
Interfaçage des ordinateurs, 712-713
Interprétation des symboles logiques, 73-74
Interrupteur bilatéral, 447-449
Interrupteur MOS, élémentaire, 431
Interruption (UCT), 727
Intervalle d'échantillonnage, 3446
Inverseur, 51-52
circuits contenant un, 53-54
Inversion, 43, 51 (*voyez* Opération NON)

L

Langages
BASIC, 721
évolués, 721
machine, 721
PASCAL, 721

Largeur d'impulsion vraie asynchrone, 187
Lecture, définition, 616
Lignes de donnée bidirectionnelles, 536, 726
Logiciel, 639
Logique à absorption de courant, 390
Logique à couplage par l'émetteur (ECL), 426-429
  circuit de base, 426-427
  caractéristiques, 428-429
  porte OU/NI, 428
Logique à injection de courant, 390
Loi associative, 63
Loi commutative, 63
Loi distributive, 63
LSI (*voyez* Intégration à grande échelle)
Lumière UV, EPROM, 633

M

Marge de bruit CA, 390
Marges de sensibilité aux bruits état HAUT, 389
Marges de sensibilité aux bruits état BAS, 389
Marges de sensibilité aux bruits, 388
  CA, 390
  CMOS, 441
  DC, 388
  NMOS, 434
Matrice de registres, 625
MBM (*voyez* Mémoire à bulles magnétiques)
Mémoire, 15
  accès sélectif, 617
  activée, 620
  amorce, 639
  bande magnétique, 615, 681-683
  bulles magnétiques, 615, 681 (*voyez* Mémoire à bulles magnétiques)
  cellule bipolaire, 653
  cellule NMOS, 653
  de masse, 618
  disque dur, 682-654
  disque magnétique, 615, 681-682
    disquette, 683
    disque dur, 682-683
    optique, 684
  extension
    capacité, 670-671
    longueur du mot, 668-669
  fonctionnement général, 618-621
  interne, 618
  magnétique, 680-684
  file, 679-680
  rémanente, 624
  terminologie, 615-617
  tore magnétique, 680-681
  volatile, 617
Mémoire à accès sélectif (*voyez* VIVES)
Mémoire à accès séquentiel (SAM), 617
Mémoire à bande magnétique, 615, 681-683
Mémoire à bulles magnétiques (MBM), 615, 681
Mémoire à disque magnétique, 615, 681-682
Mémoire à tore magnétique, 680
  tore de ferrite, 680
Mémoire de masse (*voyez* Mémoire)
Mémoire interne (*voyez* Mémoire)
Mémoire rémanente, 624, 633, 674
Mémoire temporaire, VIVE, 649
Mémoire VIVE dynamique (DRAM), 657-667
  combinaison de CI, 674
  compteur de rafraîchissement, 666
  contrôleurs, 658, 668
  cycle de lecture/écriture, 663-665
  multiplexage d'adresse, 659-661
  rafraîchissement, 657, 665-668
  structure et fonctionnement, 658-663
Mémoire VIVE intégrée, 658
Mémoire volatile, 616
Mémoires FILE, 679-680
Mémoires séquentielles, 675-680
  File (FIFO), 679-680
  registre circulaire, 676
Mémorisation et transfert des données, 194-196
Méthode de la division répétée,22
Méthode de la parité paire, 35-37
Méthode de la parité impaire, 36-37
Micro-ordinateurs, 16-17, 707-731
  applications, 203-204
  définition, 708
  éléments de base, 713-715
  exécution d'un programme en langage machine, 720-724
  instruction monoadresse, 717
  instructions à adresses multiples, 718
  instructions multioctets, 719-720
  interfaçage, 712-713
  introduction, 708-731
  microprocesseurs (*voyez* Microprocesseurs)
  mots d'instruction, 717-720
  mots machines, types, 716-717
  notion de, 708-711
  organisation de base, 711-713
  stockage du programme, 639
  structure type, 724-727
  unité d'entrée, 16, 712
  unité de mémoire, 16, 712
  unité de sortie, 16, 712
Microprocesseurs, 17, 713, 714-715
  introduction, 708-731
  opération d'écriture, 623, 729-731
  opération de lecture, 623, 727-729
  section de synchronisation et de commande, 715
  section des registres, 715
  unité arithmétique et logique (UAL), 711-712, 715
Microprogramme, 639
Miniordinateurs, 16-17
Minuterie 555 utilisée comme un multivibrateur astable, 221-222 (*voyez* Multivibrateurs astables)
Mise à zéro d'une bascule
  définition, 161
  élément de mémoire, 159
  et mise à un simultanément, 160
Mnémonique, 721
Mode de basculement, 9
Module mémoire, 669
Monostable, 206-218
  dispositifs réels, 208
Monostable non redéclenchable, 206-207
Monostable redéclenchable, 207-208
Monotonicité (d'un CNA), 572
MORTE (mémoire), 623-640
  applications, 638-640
  architecture, 625-627
  définition, 617
  grillage, 624
  opération de LECTURE d'une, 625
  programmation, 624
  programmée par masque, 628-633
  schéma fonctionnel, 624-625
  synchronisation, 627
  test, 692-693
  types de, 628-638
MORTE programmée par masque, 628-633
MORTES programmables (PROM), 632-633
MORTES programmables effaçables, 633-638
Mot d'instruction à une adresse, 717
Mot mémoire, 615
Mots d'instruction, 717-720
Mots machines, types de, 716-717
MSI (*voyez* Intégration à moyenne échelle)
Multiplexage, 496
  adresse (dans une DRAM), 659-663
  CAN, 597-599
Multiplexage d'adresse (dans une DRAM), 659-663
Multiplexé par répartition dans le temps, 515
Multiplexeur deux entrées, de base, 497
Multiplexeurs (MUX), 496-502
  applications, 502-507, 511, 662
  deux entrées, de base, 497
  huit entrées, 499
  quatre à deux entrées, 502
  quatre entrées, 498
  séquencement des opérations, dans le, 505
Multiplexeurs à 8 entrées, 499
Multiplexeurs à quatre entrées, 498
Multiplicande, 280
Multiplicateur, 280
Multiplicateur série, 281
Multiplicateurs binaires, 280-284
Multiplication
  binaire, 256-256
  ET, 49, 415
  logiciel, 284
Multiplication logique (opération ET), 43, 48-49, 415
Multivibrateur astable, 220-222
Multivibrateur monostable,218, 220 (*voyez* Monostable)
Multivibrateurs astables, 220-222
  minuterie 555 utilisée comme, 221-222
Multivibrateurs bistables, 220 (*voyez* Bascules)
MUX (*voyez* Multiplexeurs)

N

Niveau logique, 43
Niveaux logiques actifs, 73-74
Niveaux non validés, 81
Niveaux validés, 81
Nombres signés, représentation, 247-253
Notation complément à 1, 248
Notation complément à 2, 247-248
  addition et soustraction, combinée, 275-276
  multiplication, 256
  soustraction, 256
Notation fonctionnelle, 84-85
  −, 308, 325
  /, 363
  $\hat{\Omega}$, 422, 494
  ∇, 426
  &, 82, 307
  Σ (Sigma), 284-285)
  BCD/DEC, 494
  BIN/OCT, 494
  C, 325
  CI, 362-363
  CT = 0, 307
  CTR, 307
  D, 325, 362
  DIV*n*, 307-308, 325
  G, 325-326
  HPRI/BCD, 495
  Q, 524
  R, 362
  SRGn, 362
  →, 363
  ≥, 82
  ⊾, 83, 494
Notation grandeur-signe, 247
Numériser
  reconstruction d'un signal, 582-583
  signal, 573, 580-583
Numéro MODULO, 202, 297
  compteur asynchrone, 297-303
  changement du, 285-286
  compteur Johnson, 342-343
  compteur en anneau, 340-341
  variable, avec le 74412, 323-324

O

Observation/analyse, technique de dépannage, 495-496
Octet, 615
Octets, réunion, 112-113
Opérande, 711
Opération d'ÉCRITURE
  définition, 616, 725
  microprocesseur, 729-731
  mémoire VIVE, 652
Opération de lecture
  UCT, contrôlée par, 623
  définition, 616
  microprocesseur, 727-729
  VIVE, 621-652
  MORTE, 625
Opération de transfert, 194
Opération ET, 43, 48-49, 415
Opération ET câblé, 415-416
Opération NON, 43, 51-52
Opération OU câblé, 416
Ordinateurs
  micro, 16-17
  numériques, 15-17, 556 (*voyez* Micro-ordinateurs)
  principaux constituants, 16-17
  schéma fonctionnel, 6
  super, 17
  types d', 16-17
  mini, 16-17
Ordinateurs numériques, 15-17 (*voyez* Micro-ordinateurs)
Oscillateur à bascule de Schmitt, 220-221
Oscillateur régularisé par une tension (VCO), 592
Oscilloscope numérique à mémoire (ONM), 599-600

P

P (*voyez* Notation fonctionnelle)
Paires, réunion, 110-111
Paramètres de courant des circuits intégrés, 385-386
Paramètres de tension, circuit intégré, 385-386
Parité
  bit, 35-37
  générateur, 125-126
  méthode de détection des erreurs, 35-37
  vérificateur, 125-126
  vérification de la, 36-37
Partage de temps
  afficheurs, 503
  décodeurs/pilotes, 503
Pas discrets, 3
PASCAL, langage, 721
Périphériques de stockage de masse, 587
perluette
& (*voyez* notation fonctionnelle)
Pilote, décodeur, 492
Pilotes de bus, 534
Plaque arrière, ACL, 485
+ (*voyez* Notation fonctionnelle)
Point binaire, 8
Pointes, 300
  de décodage, 330-333
Pointeur d'instructions (PI), 715, 723
Porte de transmission, CMOS, 447-449
Porte ET, 49
  compteur de fréquence, 344-3449
  décodage de compteur, 326-330
  définition, 49
  description booléenne, 52-53
  évaluation de la sortie, 54-56
  matérialisation des expressions booléennes, 56-57
  représentation synonyme, 72-73
  sélection de la représentation, 76-80
  théorèmes de Boole 61-65
  récapitulation du fonctionnement
Porte flottante, EPROM, 633
Porte NI, 58-59
  bascule S-C, 163-165
  choix de la représentation, 76-82
  CMOS, 438
  définition, 58-59
  ECL, 428, 429
  élément de mémoire, 163-165
  NMOS, 434
  représentation synonyme, 72-74
  universalité de la, 68-71 (*voyez* Circuits logiques combinatoir
Porte NON-ET, 59-60
  choix de la représentation, 76-82
  circuits internes des bascules S-C déclenchées par un front, 172-173
  circuits internes des bascules J-K déclenchées par un front, 17
  CMOS, 437-438
  décodage des compteurs, 326-330
  définition, 59-60
  éléments de mémoire, 157-163, 170-173
  NMOS, 432
  représentations synonymes, 72-82
  TTL, 397-405, 411
  universalité des, 69-71 (*voyez* Circuits logiques combinatoir
Porte OU, 45-48
  choix de la représentation, 76-82
  définition, 45-46
  description booléenne, 52-54
  ECL, 428, 429
  évaluation de la sortie, 54-56
  matérialisation à partir des expressions booléennes, 56-5
  représentation synonyme, 72-82
  théorèmes de Boole, 61-65
Opération OU, 43, 44-48
Portes logiques, 41-85
  choix de la représentation, 76-82
  circuits NON (INVERSEUR), 51-5
  constantes et variables booléennes,
  description booléenne, 53
  ET, 49
  évaluation des sorties, 54-56
  matérialisation à partir d'expressio booléennes, 56-57
  NI, 58-59, 69-71
  non inverseuses, 73
  NON-ET, 59-60, 69-72
  OU, 45-48
  représentation IEEE/ANSI, 82-85 (*voyez* Symboles logiques IEEE/ANSI standard)
  représentations synonymes, 72-76
  tables de vérité, 44
  théorèmes de Boole, 61-65
  théorèmes de De Morgan, 65-68
Ports entrée/sortie, 726-727
Préréglage synchrone, 319
Problèmes de synchronisation dans les bascules, 189-191
Produit vitesse-consommation, 387-388

Programme
définition, 5, 15, 708
exécution, 722-724
Programme d'amorce, 639
Programmeurs, 708
PROM effaçable électriquement, 638-638
Puces interface entrée/sortie, 17
Puces mémoires, 17
Pulseur logique, comment s'en servir, 457-458

Q

Q (*voyez* Notation fonctionnelle)
Quartets, réunion, 111-112
4000/14000 Série CMOS, 498
4016/74HC4016 interrupteur bilatéral, 447-449
4511B Décodeur/pilote DCB-7 segments, 486
47256 (TMS), MOS, MROM, 632
4731B registres, 357
Quatre multiplexeurs deux entrées, 502

R

R (*voyez* Notation fonctionnelle)
Rafraîchissement,DRAM, 657, 665-668
RAU c.c., 181-182
RAZ c.c., 181-182
Rebondissement des contacts, 162
Recirculation des données, 675-676
Reconstitution d'un signal numérisé, 582-583
Réduction de la consommation (dans les MROM), 631
Référence de tension de précision, 568
Registre à décalage circulaire, 340-341
Registre accumulateur, 262
Registre B, 262
Registre lecture parallèle/écriture parallèle, 353-357
Registre lecture parallèle/écriture série, 358
Registre lecture série/écriture parallèle, 358
Registre lecture série/écriture série, 357
Registre tampon, 3449
Registres, 2020, 295-366
accumulateur, 262
additionneur parallèle avec, 267-269
B, 262
intégrés (*voyez* Registres intégrés)
notation, 267
transfert entre (*voyez* Transfert de données)
trois états, 528-530
Registres à décalage, 196-199
Registres intégrés, 353-362
lecture parallèle/écriture parallèle, 353-357
lecture parallèle/écriture série, 358
lecture série/écriture parallèle, 358
lecture série/écriture série, 357
Réinitialisation maîtresse, 288, 319
Remise à un d'une bascule
élément de mémoire, 158-159
et remise à zéro simultanément, 160
Remise à zéro d'une bascule, définition, 161
Report
anticipé, 271
propagation, 269-271
Report anticipé, 271
Représentation analogique, 3
Représentation d'un nombre binaire signé, 247-253
Représentation d'un nombre signé, 247-253
Représentation des nombres, 3-4
Représentation des nombres négatifs, 247-253
Représentation des nombres signés, 247-253
avec la notation en complément à 2, 248-251
Représentation numérique, 3
Représentation simplifiée des bus, 535-536
Représentations des portes laquelle utilisée, 76-82
Résolution
CAN, 579
CNA, 559-561, 571
pourcentage, 561
Résolution en pourcentage, 561
Retards de propagation ($t_{PLH/PHL}$)
compteurs asynchrones, 310-313
bascule, 186
circuit intégré, 386
porte NON-ET TTL, 399
Réunion, 110-119
octets, 112-113
paires, 110-111
quartets, 111-112
Ronds, 72-82
placement des, 78-79

S

Schéma
connexion des circuits logiques, 133-134
détermination du niveau de sortie à partir de, 55-56
transition d'état, 201-202
Schéma de transition des états, 201-202
compteur MODULO-6, pour, 285
compteur synchrone, pour, 336
décompteur MODULO-8, pour, 309
Schéma de transition, état, 201-202
Schottky TTL, série 74S, 401-402, 404
Section des registres, microprocesseur, 715
Sélecteurs de données, 496-502
Sélection de la puce, 625, 652
Séquenceur logique programmable, 648
Sigma Σ, 271, 284-285
Signaux de bus, 532
74AS, série TTL, 403
74ALS, série TTL, 403
74H, série TTL, 401, 404
74HC04, INVERSEUR,449
74HC05, INVERSEUR à collecteur ouvert, 446
74HC85 (*voyez* 74LS85)
74HC124, tampon trois états, 446
74HC160, compteur à décade synchrone, 315
74HC161, compteur MODULO-16 synchrone, 315
74HC162, compteur à décade synchrone, 315
74HC163, compteur MODULO-16 synchrone, 315
74HC164, registres, 358
74HC165, registres, 358
74HC4016, interrupteur bilatéral, 447-449
74HC4017, registres, 344
74HC4022, registres, 344
74L, série TTL, 400-401, 404
74LS, série TTL, 402-404
74LS83, additionneur parallèle 4 bits, 527
74LS85, comparateur de grandeurs 4 bits, 521-524
74LS138, décodeur/démultiplexeur, 477, 479, 510
74LS160, compteur à décade synchrone, 315
74LS161, compteur MODULO-16 synchrone, 315
74LS162, compteur à décade synchrone, 315
74LS163, compteur MODULO-16 synchrone, 315
74S, série TTL, 401-402, 404
7400, série TTL, 397-400
7442, décodeur DCB-décimal, 480
7445, décodeur/pilote DCB-décimal,481
7446, décodeurs/pilotes,484
7447, décodeurs/pilotes, 484
7485, comparateur de grandeurs 4 bits, 521-524
74147, codeur de priorité décimal-DCB, 488-502
74151, multiplexeur huit entrées, 499, 504, 507, 511
74157, quatre multiplexeurs deux entrées, 502
74164, registres, 358
74165, registres, 358
74173, registres trois états, 529-531
74174 , registres, 353
74178, registres, 353-356
74193 (LS193/HC193), compteur, 319-324
74194 (LS194/HC194), registre à décalage bidirectionnel, 361
74293, compteur, 481
74373 (LS373/HC373/HCT373), registre octal 8 bits, 362
74374 (LS374/HC374/HCT374), registre octal 8 bits, 362
Somme de contrôle, 692
Sonde logique, comment s'en servir, 134-143, 457
Sortance, 386
CMOS, 442
détermination de la, 406
MOS, 434
TTL, 405-411
Tore de ferrite, 680
Sortie de bascule normale, 156
Sortie de retenue, 320
Sortie inversée d'une bascule, 156
Sortie pleine échelle (d'un CNA), 559
Sortie report,319
Sorties à collecteur ouvert, 399
décodeurs, 481
TTL, 415-422
Sorties de comptage résultantes, 319
Sorties de décomptage résultantes, 319-320
Sorties du nombre, 319-218
Sorties multiples, circuits logiques avec, 1128-129
Sorties prioritaires, 181
Soustraction
dans la notation en complément à 2, 254-256, 273-275
hexadécimale, 260-262
Soustraction hexadécimale, 260-262
SRG*n* (*voyez* Notation fonctionnelle)
SSI, 129
Strobage, 332
Strobe d'adresse de colonne (CAS), 659-661
Strobe d'adresse de rangée (RAS), 659-661
Substrat, 129
Superordinateurs, 17
Symboles logiques standard IEEE/ANSI, 82-85, 183-185, 284-285, 516-518, 524
6264 8 K X 8 SRAM, 657
74147 HPRI/BCD, 495
74150 MUX, 516
74151 MUX, 516
74157 MUX, 516
74164 SRG8, 363
74174 Bascules, 363
74193 compteur, 324
74293 compteur, 307
7442 BDC/DEC, 494
7445 BDC/DEC, 494
74HC4024 compteur à propagation, 308
74LS138 BIN/OCT, 494
74LS138 DEMUX, 518
74LS85 COMP, 524
additionneur complet, 285
additionneur parallèle, 4 bits, 285
bascule à mémoire D, 184
bascule D, 185
bascule J-K, 184
bascules, 183-185
bloc de commande commun, 185, 307, 362
CI complexes, 84-85
comparateur de grandeurs, 524
définition, 82-83
ET, 83
inverseur, 83
monostable, 208
multivibrateur monostable,208
NI, 83
NI exclusif, 121
NON-ET, 83
notation fonctionnelle
OU, 83
OU exclusif, 120
pilote, 494
sens du comptage (+ ou −), 325
sortie à collecteur ouvert, 422
sorties trois états, 426
tampon, 494
Symboles rectangles (*voyez* Symboles logiques IEEE/ANSI standard)
Synchronisation, bascule, 191-193
Système à base de 10, 7
Système à poids positionnels, 7
Système binaire, 8-9
comparaison avec DCB, 30
conversion binaire-décimal, 21
conversion binaire-hexadécimal, 28
conversion binaire-octal, 24, 475-476
conversion décimal-binaire, 21-22
conversion hexadécimal-binaire, 27-28
conversion octal-binaire, 24
forme complément pour binaire, 251
représentation, 10-11
transmission parallèle et série, 13-14
Système de bus, 724-727
Système de numération hexadécimale, 25-29
Système de numération octal, 22-25
comptage, 24
utilité du, 24
Système de télésurveillance, 511-513
Système de transmission de données synchrone, 513-516
fonctionnement complet, 515-516
multiplexage par répartition dans le temps, 515
récepteur, 513
Système décimal, 7
conversion binaire-décimal, 21
conversion décimal-binaire, 21-22
conversion décimal-hexadécimal, 27
conversion décimal-octal, 23-24
conversion hexadécimal-décimal, 26
conversion octal-décimal, 23
Système hybride, 6
Systèmes à disquettes, 683
Systèmes asynchrones, 167
Systèmes de numération, 6-11, 19-37
binaire, 8-9 (*voyez* Système binaire)
codes, 29-34 (*voyez* Codes binaires)
décimal, 7 (*voyez* Système décimal)
hexadécimal, 26-29
octal, 22-25
Systèmes logiques séquentiels, dépannage, 363
Systèmes numériques, 4
dépannage, 134-143
alimentation défectueuse, 141
conducteurs de signaux court-circuités, 140
conducteurs de signaux ouverts, 139-140
court-circuit entre deux broches, 138-139
dysfonctionnement dans les circuits internes, 135-136
entrée court-circuitée à la masse ou à l'alimentation par l'intérieur, 136
entrée ou sortie en circuit ouvert, 137
étude de cas, 141-143
sortie court-circuitée à la masse ou à l'alimentation par l'intérieur, 136
notions de base, 1-17
Systèmes synchrones, 167-169

T

Table
d'états, 200-201
d'excitation de circuit, 337
d'excitation J-K, 334
Table des états, 200-201
Tables de données, 639
Tables de vérité, 44
Tampon/pilote, 420-421
Tampon trois états inverseurs, 424-425
Tampons, 357
collecteur ouvert, 420-421
non inverseur, 424
sortie, MORTE, 627
trois états, 424-425
Tampons de sortie, MORTE, 627
Tampons/pilotes à collecteur ouvert, 401-422
Techniques numériques
avantages, 5
limites, 6
Temps d'accès, 696
définition, 616
MORTE, 627
Temps d'acquisition, circuit échantillonnage et mémorisation, 597
Temps d'établissement, d'un CNA, 571-572
Temps de maintien (t_M), 169, 186, 197
Temps de stabilisation (t_s), 169, 186
Temps de validation de la sortie, 627
Tension de décalage, d'un CNA, 572
Tension seuil descendante (V_{T-}), 206
Tension seuil montante (V_{T+}), 201
Tensions indéterminées, 132-133
Test automatique des circuits (avec CAN), 573
Test d'exactitude statique, d'un CNA, 573-574
Test en escalier, d'un CNA, 573-574
Tête de lecture/écriture, 682
Théorèmes
de Boole, 61-64
de De Morgan, 65-68
plusieurs variables, 63-64
Théorèmes de Boole, 61-65
Théorèmes de De Morgan, 65-68
conséquences, 67-68
tiret
− (*voyez* Notation fonctionnelle)

Traceur galvanométrique, 143, 457-458
Transducteur, 555
Transfert asynchrone, 194
Transfert de données en série, 196-199
  entre registres, 197-194
Transfert de données parallèle, 177-178, 191-196
  comparé au transfert série, 199
Transfert des données, 194-196
  asynchrone, 194
  démultiplexeurs, 508-516
  et mémorisation, 194-196
  opération, 532
  par bus de donnée, 530-537
  parallèle, 177-178, 191-196
  parallèle comparé au transfert série, 199
  registres à décalage, 196-199
  registres, entre, 532
  série, 196-199
    exigence du temps de maintien, 197
  simultané, 191
  synchrone, 194
Transfert parallèle, 191, 317
Transfert synchrone, 194
Transistor à consommation de courant, TTL, 395
Transistor à injection de courant, TTL, 396
Transistor d'entrée à émetteurs multiples, 393
Transistor de rappel à la source, TTL, 396
Transistor de rappel à la masse, TTL, 395
Transistor MOS, 427-436
Transistor MOS à appauvrissement, 430
Transistor MOS à enrichissement, 430
Transition descendante (FD), 168
Transmission en série, 13-14
Transmission parallèle, 13-14
Transparent (élément de mémoire D), 179
Transposeur de niveau de tension, 452, 456
Trois états,
  bus de donnée, 425
  CI, 425-426
  registres, 528-530
  tampons,424-425
TTL faible consommation, série 74L, 400-401, 404
TTL rapides, série 74H, 401, 404
TTL Schottky avancé, série 74AS (AS-TTL), 403
TTL Schottky faible consommation, série 74LS (LS-TTL), 402-404
TTL Schottky faible consommation avancé, série 74ALS (ALS-TTL), 403
TTL trois états, 403-426
  avantages des, 424
Tube Nixie, 330

U

ULSI, 129
1 (*voyez* Notation fonctionnelle)
Unité arithmétique et logique (UAL), 16, 262, 284 711-712, 715
Unité centrale de traitement (UCT), 713 (*voyez* Microprocesseurs)
Unité d'entrée, 16, 712
Unité de commande, 16, 713
Unité de mémoire, 16, 712
Unité de sortie, 16, 712
Unité de traitement arithmétique, 284
Universalité des portes NON-ET et des portes NI, 68-71

V

Verrouillage des données, 191
2147, symbole logique (4K × 1 VIVE), 652
2168, symbole logique (4K × VIVE), 652
2732, (4K × 8 EPROM), 636
Virgule décimale, 7
Virgule octale, 23
VIVE rémanente (NVRAM), 674-675
VIVE statique (SRAM), 652-657
  cycle d'écriture, 655
  cycle de lecture, 654-655
  puce réelle, 656
  synchronisation, 653-654
VIVES (mémoires)
  à semiconducteurs, 649
  architecture, 649-652
  autotest au démarrage, 691
  définition, 6250
  dispositifs dynamiques, 617
  extension de la capacité, 670-674
  extension de la longueur du mot, 668-670
  rémanente (NVRAM), 674-675
  statique (SRAM), 652-657
VLSI, 129
Voltmètre numérique, 593-596